

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 – Комп'ютерні науки
Освітньо-професійна програма «Управління проєктами»

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему:

**«Дослідження процесів управління проєктом створення веб-
платформи Workgram»**

Студентки 2-го курсу групи УПЗ-21

Науковий керівник:

Марії ЦИБЕНКО

(ім'я, прізвище)

К.Т.Н., доцент

(науковий ступінь, вчене звання)

Богдан ЄРЕМЕНКО

(ім'я, прізвище)

(підпис студента)

(дата)

(підпис)

Попередній захист:

(Висновок: "До захисту в Екзаменаційній комісії")

Завідувач кафедри
технологій управління

Віктор МОРОЗОВ

(підпис)

(ім'я, прізвище)

(дата)

Київ – 2025

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Освітній рівень Магістр

Спеціальність 122 Комп'ютерні науки

Освітньо-професійна програма Управління проєктами

ЗАТВЕРДЖУЮ

Завідувач кафедри

професор Віктор МОРОЗОВ

«27» вересня 2025 року

**ЗАВДАННЯ
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студентка: Марія Цибенко

Група: УПз-21

1. Тема кваліфікаційної роботи: «Дослідження процесів управління проєктом створення веб-платформи Workgram»

Затверджена Протоколом №15 від 16.06.2025 року.

2. Строк подання студентом готової роботи – «10» 12.2025 р.

3. Цільова установка та вихідні дані до роботи: дослідження методів і інструментів управління проєктом створення веб-платформи для майбутніх робітників та роботодавців, що поєднує функціонал соціальної мережі й сервісу пошуку роботи. У роботі розглянуто вибір і обґрунтування використання гнучкої методології Scrum для організації команди та планування ітерацій. У межах дослідження сформовано концептуальну та математичну моделі платформи, розроблено архітектуру програмного забезпечення та структуру бази даних, визначено методи планування, управління ризиками та забезпечення якості. На основі аналізу ринку цифрових HR-рішень і конкурентного середовища визначено ключові тенденції, потреби користувачів, проблеми молодих пошукачів та роботодавців. Сформовано функціональні та нефункціональні вимоги до

системи, побудовано структурну декомпозицію робіт (WBS) та організаційну структуру Scrum-команди.

4. Зміст роботи: обґрунтування актуальності і доцільності створення веб-платформи, яка забезпечує сучасний, зручний та візуально орієнтований процес пошуку роботи, подібний до соціальних мереж. Проведено комплексний аналіз ринку цифрових HR-технологій, конкурентного середовища та зацікавлених сторін. Виконано PEST- та SWOT-аналіз.

Розроблено концептуальну модель інформаційної системи та формалізовано математичну модель предметної області з використанням множин, відображень і функцій, що описують взаємодію між сутностями: користувачами, роботодавцями, вакансіями, відгуками й рекомендаційними механізмами.

Визначено функціональні та нефункціональні вимоги до платформи. Обґрунтовано вибору методології, розроблено WBS, організаційну структуру команди та беклог продукту.

Проведено ідентифікацію ризиків, здійснено їх кількісну та якісну оцінку. Виконано розрахунок загального бюджету на створення та підтримку платформи.

5. Перелік графічного матеріалу:

Дерево проблем і дерево цілей; PEST- та SWOT-аналіз; мапа стейкхолдерів; концептуальна модель інформаційної системи; архітектура програмного забезпечення; організаційна структура Scrum-команди; WBS та контрольні віхи; беклог продукту і спринти; концептуальна, логічна та фізична моделі бази даних; ресурси й бюджет проєкту; карта ризиків; дизайн-система та макети інтерфейсів платформи.

6. Календарний план виконання роботи

№ з/п	Назва частин роботи	Виконання роботи
1	Вивчення літературних джерел з предмету дослідження	01.10.25-11.10.25
2	Збір і вивчення матеріалів	12.10.25-19.10.25

3	Складання розгорнутого плану кваліфікаційної роботи	20.10.25-23.10.25
4	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін	24.10.25-25.10.25
5	Підготовка розділу 1	26.10.25-02.11.25
6	Підготовка розділу 2	03.11.24-14.03.24
7	Підготовка розділу 3	11.11.25-17.11.25
8	Підготовка розділу 4	18.11.25-05.12.25
9	Оформлення кваліфікаційної роботи	25.11.25-02.12.25
10	Передача кваліфікаційної роботи науковому керівникові	03.12.25
11	Попередній захист кваліфікаційної роботи	10.12.25-14.12.25
12	Передача кваліфікаційної роботи рецензенту для рецензування	15.12.25

Дата видачі завдання «30» вересня 2025 р.

Керівник роботи доцент Богдан ЄРЕМЕНКО

(посада, ім'я, прізвище)

(підпис)

Завдання прийняла до виконання студентка групи УПз-21

Марія ЦИБЕНКО

(ім'я, прізвище)

(підпис)

АНОТАЦІЯ

кваліфікаційної роботи магістра на тему

«Дослідження моделей управління проектом створення веб-платформи для об'єднання ініціатив із післявоєнної відбудови України»

Студентки: Цибенко Марії Володимирівни

Науковий керівник: Єременко Богдан Михайлович

Рік захисту — 2025

Мета кваліфікаційної роботи полягає у дослідженні та впровадженні ефективних моделей управління IT-проектом для створення веб-платформи пошуку роботи та взаємодії між роботодавцями й майбутніми працівниками, яка за принципами функціонування та інтерфейсом нагадує соціальну мережу Instagram.

Ціль проекту – створення інтерактивної платформи, що об'єднує публікацію вакансій у вигляді постів, ведення профілів роботодавців та пошукачів, стрічку рекомендацій, систему миттєвих повідомлень, можливість відгуків та коментарів, а також інструменти аналітики для роботодавців. Платформа має забезпечити швидку й зручну комунікацію між двома сторонами ринку праці, спростити пошук роботи та покращити якість рекрутингу.

Практична цінність роботи полягає у створенні цифрового рішення, яке поєднує механіку соціальної мережі зі структурованими інструментами HR-процесів. Платформа забезпечує можливості для публікації вакансій, взаємодії користувачів через лайки, коментарі та чат, застосування фільтрів пошуку, автоматизацію збору відгуків та аналітики. Сервіс створює єдину інформаційну екосистему для роботодавців і здобувачів, підвищує прозорість комунікацій та сприяє оперативному працевлаштуванню.

Кваліфікаційна робота складається з анотації, вступу, основної частини (чотири розділи), висновків та переліку використаних джерел.

Об'єктом дослідження є процеси управління IT-проектами зі створення веб-платформ у сфері цифрових HR-рішень.

Предметом дослідження є методи, моделі та інструменти управління проектом розробки веб-платформи для взаємодії майбутніх працівників і роботодавців із використанням гнучких методологій управління.

У роботі досліджено сучасні підходи до управління IT-проєктами та обґрунтовано доцільність застосування методології Agile/Scrum для розробки веб-платформи з динамічними вимогами та орієнтацією на користувацький досвід. Проведено аналіз ринку цифрових HR-платформ, визначено потреби цільової аудиторії та сформовано функціональні й нефункціональні вимоги до системи.

Розроблено концептуальну модель інформаційної системи, архітектуру програмного забезпечення та структуру бази даних. Побудовано ієрархічну структуру робіт (WBS), сформовано беклог продукту та організаційну структуру Scrum-команди. Проведено ідентифікацію та оцінку ризиків, розроблено протиризикові заходи та виконано розрахунок бюджету проєкту.

Отримані результати можуть бути використані під час планування та реалізації веб-платформ у сфері HRTech і цифрових сервісів працевлаштування.

Робота завершується висновками щодо досягнення поставлених завдань та формує рекомендації з подальшого розвитку платформи, включаючи майбутню інтеграцію з HR-системами, додавання алгоритмів рекомендацій вакансій і масштабування сервісу.

Кваліфікаційна робота складається з 90 сторінок основного тексту, містить 11 рисунки, 36 таблиці, 5 формул та 4 додатки.

Ключові слова: управління проєктами, Agile, Scrum, веб-платформа, соціальна мережа, вакансії, стрічка постів, роботодавці, пошукачі, онлайн-чат, прототипування, управління ризиками.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ ТА ЖИТТЄЗДАТНОСТІ ПРОЄКТУ	12
1.1 Аналіз ринку працевлаштування і наявних платформ.....	12
1.2 Проблеми взаємодії роботодавців та кандидатів в онлайн середовищі	14
1.3 Огляд аналогів	23
1.4 Характеристика цільового користувача	26
1.5 Постановка задачі, формулювання ТЗ, паспорт проєкту	27
РОЗДІЛ 2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА МОДЕЛЮВАННЯ СИСТЕМИ	30
2.1 Побудова концептуальної моделі системи	30
2.2 Формалізація завдань пошуку та підбору кандидатів	33
2.3 Математичне моделювання фільтрації вакансій і кандидатів	37
2.4 Основи використання рейтингових алгоритмів	40
РОЗДІЛ 3. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ	46
3.1 Складання функціональних і нефункціональних вимог.....	46
3.2 Структурна декомпозиція робіт та організаційна структура проєкту	53
3.3 Use Case діаграми (реєстрація, пошук, чат, вакансії тощо)	58
3.4 Управління ризиками	61
3.5 Бюджет проєкту	64
РОЗДІЛ 4. ПРОГРАМНА РЕАЛІЗАЦІЯ	71
4.1 Архітектура веб-платформи	71
4.3 Тестування окремих функціональних елементів інформаційної системи	82
4.4 Управління якістю.....	85
ВИСНОВОКИ	90
ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ	92
ДОДАТКИ	95
Додаток А.....	95
Додаток Б	98

Додаток В	101
Додаток Г	106

ВСТУП

У ХХІ столітті цифровізація стала ключовим фактором розвитку економіки, освіти та ринку праці. Глобальні технологічні зміни, поширення дистанційної роботи, трансформація способів комунікації та впровадження алгоритмічних систем управління значно змінили механізми взаємодії між роботодавцями та здобувачами. У таких умовах питання ефективного пошуку роботи, стажувань і перших професійних можливостей набуває особливої актуальності, оскільки сучасна молодь стикається з низкою викликів, які не завжди враховуються традиційними підходами до працевлаштування.

Серед ключових викликів для студентів та молодих фахівців можна виокремити складність адаптації до вимог ринку праці, недостатність досвіду, обмеженість проміжних можливостей (стажування, практика, тимчасова зайнятість), невміння презентувати власні компетенції та навички, а також інформаційну асиметрію між учасниками ринку праці. Традиційні веб-ресурси з пошуку роботи здебільшого орієнтовані на досвідчених фахівців та не забезпечують персоналізованого підходу, адаптованого до потреб молоді. На цьому тлі зростає потреба у розробці спеціалізованих цифрових рішень, здатних надати молодим людям структуровану, зрозумілу та інтерактивну платформу для пошуку можливостей, формування професійного портфолію та взаємодії з роботодавцями.

Водночас роботодавці також стикаються із проблемами в роботі з молодими спеціалістами: складністю оцінки їхнього потенціалу, недостатністю інформації про реальні навички кандидатів, низькою мотивацією або нерелевантністю поданих резюме. Багато компаній зацікавлені у формуванні власного кадрового резерву та залученні студентів, проте існуючі інструменти не дають змоги ефективно організувати цей процес. У цих умовах цифрові платформи можуть виконувати роль посередника нового типу — гнучкого, адаптивного та орієнтованого на конкретну аудиторію.

Сучасні тренди розвитку цифрових сервісів демонструють, що молодіжна аудиторія надає перевагу інтерфейсам, близьким до соціальних мереж, мобільності, візуальності, швидкій інтерактивності та простоті використання. Платформи, які виглядають як Instagram, TikTok або Telegram, сприймаються молоддю інтуїтивно, що значно підвищує ефективність їх використання. Тому розробка платформ працевлаштування нового покоління, оформлених у стилі соціальних мереж, із функціоналом рекомендацій, картковими інтерфейсами, чатами та швидкою взаємодією, стає актуальним напрямом не лише IT-індустрії, а й освітньої та кадрової політики.

У зв'язку з цим виникає потреба у створенні веб-платформи, яка не тільки з'єднує роботодавців і молодих спеціалістів, а й забезпечує комплексний підхід до формування та підтримки освітньо-професійної траєкторії. Така система має виконувати функцію цифрового кар'єрного помічника, що враховує навички, інтереси, мотивацію та індивідуальні особливості пошукача. Паралельно платформа повинна надавати роботодавцям зручні інструменти для аналізу кандидатів, комунікації та формування внутрішніх молодіжних програм.

З огляду на зазначене, розробка веб-платформи для працевлаштування студентів та молоді, інтегрованої з інструментами рекомендаційних систем, UX/UI-підходами соціальних мереж та елементами управління проєктами (WBS, процесний аналіз, моделювання), є на часі. Такий продукт може бути застосований як інструмент розвитку кар'єрної підтримки в освітніх закладах, як основа для стартапу або як внутрішній корпоративний сервіс для відбору молодих кадрів.

Мета роботи — розробити проєкт веб-платформи, яка буде об'єднувати роботодавців і молодих спеціалістів/студентів, сприяючи ефективному підбору вакансій, стажувань та практик.

Об'єкт дослідження: процес пошуку та підбору персоналу в цифровому середовищі.

Предмет дослідження: методи, моделі та технології проектування інформаційної веб-платформи як засобу взаємодії між шукачами й роботодавцями.

Методи дослідження: аналіз, моделювання, WBS-структурування, CASE-методи, MS Project, UML.

Практичне значення: розробка структури та функціоналу платформи, що може бути реалізована як MVP або стартап для навчального закладу чи кар'єрного центру.

Завдання:

- Обґрунтувати актуальність створення нової платформи для працевлаштування в умовах цифровізації та зростання попиту на візуальні сервіси.
- Розробити концептуальну модель веб-платформи з описом основних підсистем і взаємодії між користувачами.
- Побудувати формальну (математичну) модель системи у вигляді множин, функцій та відношень.
- Визначити функціональні та нефункціональні вимоги до платформи, орієнтованої на молодіжну аудиторію.
- Спроекувати користувацький інтерфейс, натхненний соціальними мережами, з урахуванням принципів UX/UI-дизайну.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ ТА ЖИТТЄЗДАТНОСТІ ПРОЄКТУ

1.1 Аналіз ринку працевлаштування і наявних платформ

Стрімкий розвиток цифрових технологій зумовив кардинальні зміни у способах пошуку роботи та підбору персоналу. Веб-платформи стали ключовим інструментом взаємодії між роботодавцями та кандидатами, витіснивши традиційні підходи, такі як друковані оголошення чи особисті візити до компаній. Згідно зі світовими дослідженнями, понад 85 % усіх вакансій на сучасному ринку розміщується в онлайн-середовищі, а кількість користувачів професійних платформ зростає щороку.

Ринок працевлаштування в Україні та світі активно розвивається у напрямку цифровізації. Веб-платформи відіграють ключову роль у пошуку роботи, а також у формуванні іміджу компанії як роботодавця. Серед найпопулярніших сервісів працевлаштування в Україні можна виділити: Work.ua, Rabota.ua, Jooble, Djinni, а також міжнародні платформи — LinkedIn, Indeed, Glassdoor[12, 13, 14].

Платформи відрізняються за типами вакансій, які вони пропонують, фокусом на аудиторію, рівнем автоматизації процесів пошуку та підбору персоналу. Наприклад:

- Work.ua — масова платформа, орієнтована на широкий сегмент користувачів, проста у використанні, однак не має інструментів для молодіжного сегмента (стажування, практика)[12];
- Rabota.ua — подібна до Work.ua, орієнтована на різні галузі, має зручні фільтри, однак функціонал для студентів обмежений[13];
- Jooble — агрегатор вакансій з інших платформ, без власної бази користувачів[14];
- LinkedIn — міжнародна професійна мережа, сильна в B2B-сегменті, але складна для новачків[15];
- Djinni — спеціалізована платформа для IT-сфери з анонімною взаємодією[16].

Тобто сучасні платформи працевлаштування можна поділити на декілька основних категорій:

- універсальні платформи (Indeed, LinkedIn, Jooble), що охоплюють широкий спектр професій та галузей;
- спеціалізовані платформи, орієнтовані на конкретні професії (Behance для дизайнерів, Djinni для IT-фахівців);
- корпоративні системи підбору кадрів, інтегровані у внутрішні HR-процеси організацій;
- молодіжні та студентські платформи, що надають стажування, практику та вакансії для початківців.

Попри розвиток технологій, актуальними залишаються проблеми інформаційного перевантаження, низької релевантності вакансій, недостатньої персоналізації контенту, а також нерівномірної доступності платформ для різних груп користувачів. Саме це формує потребу у створенні нових веб-платформ, що орієнтовані на молодь, мають покращену функціональність, сучасний дизайн та алгоритми рекомендацій.

Аналіз вказує на те, що на українському ринку майже відсутні інструменти, спеціально спрямовані на підтримку молодих фахівців без досвіду, студентів, які шукають перше місце роботи, стажування або практику. Крім того, більшість сервісів обмежуються формальним пошуком за ключовими словами без застосування рекомендаційних систем чи інструментів адаптивного підбору за навичками, інтересами та профілем користувача.

Іншою проблемою є відсутність інтеграції з освітніми установами або центрами кар'єри, що могло б забезпечити релевантність вакансій до напрямів підготовки студентів. Таким чином, існує реальна потреба у створенні нової платформи, яка б не лише сприяла пошуку роботи, а й підтримувала комунікацію між вищими навчальними закладами, роботодавцями та молоддю.

Запропонований проєкт веб-платформи покликаний закрити цю нішу та створити зручне, функціональне і безпечне середовище для професійного старту молоді.



Рис. 1.1 Кількість вакансій відповідно до аналітики команди [Robota.ua](https://roboota.ua)

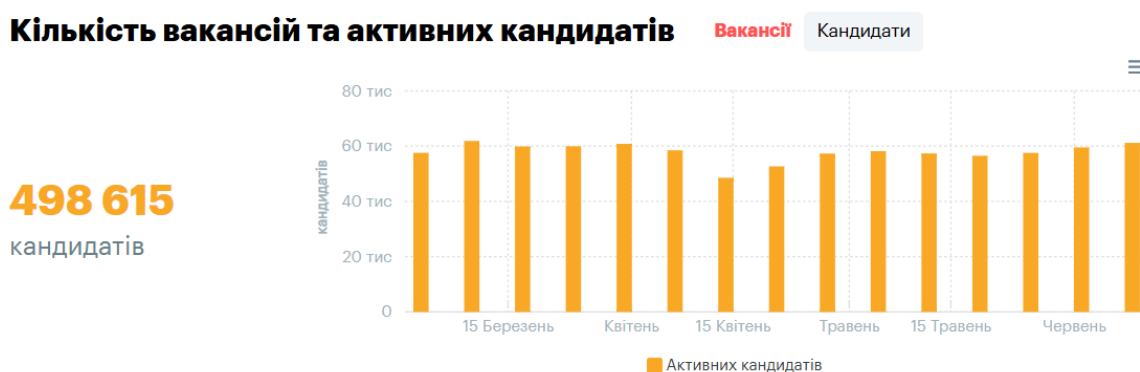


Рис. 1.2 Кількість активних кандидатів відповідно до аналітики команди [Robota.ua](https://roboota.ua)

1.2 Проблеми взаємодії роботодавців та кандидатів в онлайн середовищі

Розвиток цифрових технологій докорінно змінив механізми взаємодії між роботодавцями та кандидатами на ринку праці. Онлайн-платформи стали основним інструментом для пошуку роботи, стажувань, практик та професійних можливостей. Проте, незважаючи на стрімке зростання цифрових сервісів, взаємодія між учасниками ринку праці в онлайн-середовищі досі залишається недостатньо ефективною та вимагає удосконалення. Це стосується особливо молоді, для якої цифровий спосіб комунікації є базовим, але при цьому відсутність досвіду ускладнює проходження всіх етапів підбору персоналу.

Попри наявність великої кількості онлайн-ресурсів для пошуку роботи, взаємодія між роботодавцями та здобувачами досі стикається з низкою проблем, які ускладнюють процес працевлаштування, особливо для молоді.

Проблематика взаємодії між роботодавцями та кандидатами багатогранна й охоплює соціальні, психологічні, комунікативні, технологічні та організаційні аспекти. Нижче наведено комплексний аналіз ключових бар'єрів, що гальмують ефективну комунікацію на ринку праці:

1. Інформаційна асиметрія та непрозорість процесів

Однією з головних структурних проблем ринку праці є інформаційна асиметрія — нерівномірний розподіл інформації між шукачами та роботодавцями. Часто роботодавець має більше відомостей про вимоги, очікування та корпоративну культуру, ніж кандидат. У свою чергу, кандидат не завжди може якісно продемонструвати свій потенціал, оскільки не володіє достатнім досвідом або знаннями про специфіку певної сфери.

Для молодих спеціалістів проблема ускладнюється низкою чинників:

- відсутність реального досвіду роботи;
- нестача практичних навичок у підготовці резюме;
- нерозуміння неформальних правил підбору персоналу;
- труднощі у самооцінюванні професійних компетенцій.

У результаті виникає ситуація, коли кандидат має значний потенціал, але роботодавець не бачить його через слабку комунікацію або недостатність інформації.

2. Низька якість комунікації та зворотного зв'язку

Однією з найбільш поширених проблем, які відзначають здобувачі, є відсутність якісного зворотного зв'язку. На популярних платформах на одну вакансію може приходити від 100 до 500 і більше відгуків, що значно ускладнює комунікацію з боку роботодавця.

Типові труднощі включають:

- відсутність відповіді після подачі заявки;
- довгі терміни розгляду;

- неясні критерії відбору;
- автоматизовані формальні відписки;
- неможливість задати уточнювальні запитання.

Це створює відчуття невизначеності, знижує мотивацію та погіршує досвід взаємодії з платформою. Для студентів це стає додатковим бар'єром входження на ринок праці.

3. Розрив між очікуваннями роботодавців і кандидатів

Невідповідність очікувань є результатом складної взаємодії кількох факторів:

- нечіткі описи вакансій або занадто загальні формулювання;
- приховані вимоги (наприклад, очікування досвіду, якого немає у молоді);
- невказана заробітна плата, що ускладнює ухвалення рішень;
- недостатність інформації про проєкти, команди, умови.

Такі обставини часто призводять до того, що кандидат відгукується на нерелевантні вакансії, а роботодавець отримує невідповідні резюме. Це збільшує конкуренцію та знижує ефективність добору персоналу.

4. Проблеми контенту резюме та формального відбору

Більшість платформ використовує механізм пошуку, побудований на основі ключових слів у резюме, що створює низку проблем:

- кандидати з нестандартним описом навичок не потрапляють у вибірку;
- пошук не враховує soft skills та потенціал до навчання;
- алгоритми не розпізнають портфоліо, творчі проєкти тощо;
- молодь часто оформлює резюме неправильно або непрофесійно.

Це призводить до формального, поверхневого аналізу кандидатів, який не відображає їхні реальні можливості.

5. Проблеми довіри та безпеки

У цифровому середовищі найпоширенішими ризиками є:

- фейкові вакансії;
- шахрайські оголошення;

- недостовірні профілі кандидатів;
- відсутність верифікації роботодавців;
- продаж особистих даних третім особам.

Недовіра до цифрових платформ значно знижує їх ефективність. Для студентів та молоді, які часто не мають досвіду розпізнавання шахрайських схем, ці ризики є особливо високими.

6. Недостатня адаптація платформ для молодіжної аудиторії

Популярні платформи орієнтовані на масовий ринок і не враховують специфіку:

- пошуку першої роботи;
- стажувань та практик;
- навчальних проєктів;
- тимчасової зайнятості;
- гнучких форматів (онлайн-інтернатури, волонтерства).

Молоді користувачі часто не можуть знайти релевантні можливості через відсутність фільтрів або розділів, адаптованих до їхніх потреб.

Також більшість сервісів не мають інтерфейсу, близького до форматів соціальних мереж, якими користується молодь щоденно.

7. Відсутність інтеграції з навчальними закладами

Освітні установи відіграють ключову роль у формуванні професійної траєкторії студентів, проте:

- університети рідко мають власні сучасні цифрові кар'єрні центри;
- інформація про практики розміщується на різних сайтах і не зводиться до єдиної системи;
- роботодавці не мають централізованого доступу до студентських профілів;
- студенти не бачать актуальних можливостей у реальному часі.

Через це виникає дисонанс між потребами ринку та освітнім процесом.

8. Проблеми релевантності рекомендацій та фільтрації

Навіть сучасні платформи нерідко пропонують нерелевантні вакансії через:

- низьку точність алгоритмів;
- опору лише на ключові слова;
- нечіткі критерії релевантності;
- відсутність персонального аналізу поведінки користувача.

Молодь часто витрачає годинами, сортуваючи вакансії, більшість з яких не відповідає їхньому досвіду або інтересам.

9. Конкуренція та втрата видимості кандидатів-початківців

На популярних платформах:

- вакансії отримують сотні відгуків;
- профілі досвідчених кандидатів витісняють молодих;
- пошукачі без досвіду майже не мають шансів потрапити в топ-список.

У результаті роботодавці просто не бачать перспективних молодих кандидатів, навіть якщо вони мають високий потенціал.

10. Якість профілів кандидатів

Недосвідчені пошукачі часто:

- не вміють презентувати soft та hard skills;
- не мають портфоліо;
- не можуть описати досягнення в навчанні або проєктній діяльності;
- використовують шаблонні резюме, не адаптовані під вакансію.

Це суттєво знижує їхню конкурентоспроможність.

11. Проблеми комунікації та тривалий час обробки заявок

Більшість кандидатів скаржиться на:

- дуже довгі терміни відповіді (від тижнів до місяців);
- відсутність можливості швидко уточнити інформацію;
- неможливість простежити статус заявки;
- відсутність реального часу комунікації.

Для молоді, яка звикла до динамічної комунікації у соцмережах, такі процеси є демотивуючими.

Усі наведені проблеми демонструють комплексність викликів, що стоять перед роботодавцями і кандидатами в онлайн-середовищі. Сучасні платформи:

- не враховують потреби молоді,
- не забезпечують інтерактивності,
- мають низьку персоналізацію,
- створюють значний інформаційний шум,
- не гарантують довіри і безпеки.

Це підкреслює актуальність створення нової інноваційної платформи, яка:

- забезпечуватиме прозору та зручну комунікацію;
- пропонуватиме адаптивні рекомендаційні алгоритми;
- верифікуватиме дані роботодавців і кандидатів;
- дозволятиме створювати візуальні професійні портфоліо;
- інтегруватиметься з навчальними закладами;
- матиме інтерфейс, зрозумілий молодому поколінню;
- підтримуватиме швидкі канали комунікації (чат, push-сповіщення).

Ці фактори вказують на необхідність створення інноваційної платформи, яка забезпечить ефективну, прозору й адаптовану до потреб обох сторін взаємодію, з акцентом на довіру, простоту користування та орієнтацію на молодіжний сегмент.

Аналіз дерева проблем і дерева цілей проєкту

Дерево проблем є інструментом системного аналізу та візуалізації, який дозволяє чітко ідентифікувати центральну проблему проєкту, її першопричини та негативні наслідки. Такий підхід забезпечує логічне обґрунтування необхідності створення інформаційної системи та слугує основою для формування цілей проєкту. Аналіз розпочинається з

визначення ключового негативного стану, який потребує вирішення, після чого виконується його декомпозиція за рівнями.

Структура дерева проблем проєкту веб-платформи Workgram складається з трьох основних рівнів:

- Рівень 1 (коріння) — глибинні причини виникнення проблем;
- Рівень 2 (стовбур) — центральна проблема, на яку спрямовано проєкт;
- Рівень 3 (гілки) — негативні наслідки для користувачів і ринку праці загалом.

Елементи дерева проблем проєкту подано на рис. 1.3



Рис 1.3 Дерево проблем

Візуалізована схема демонструє причинно-наслідкові зв'язки, що обґрунтовують необхідність створення веб-платформи для взаємодії студентів, молодих фахівців і роботодавців. Центральною проблемою визначено неефективну взаємодію між пошукачами роботи та роботодавцями в умовах фрагментованого ринку цифрових HR-сервісів.

Формування цієї проблеми зумовлене кількома групами фундаментальних причин.

По-перше, спостерігається розпорошеність інформації про вакансії та кандидатів. Дані розміщені на різних платформах, що унеможлиблює формування єдиного профілю користувача та ускладнює процес аналізу навичок, досвіду й кар'єрних цілей молодих фахівців.

По-друге, існує низька зручність і застарілі підходи до пошуку роботи, які не відповідають звичкам цифрового покоління. Більшість платформ мають перевантажені інтерфейси, складні форми відгуків та відсутність інтерактивної стрічки, що знижує залученість користувачів.

По-третє, проблемою є обмежена персоналізація та відсутність ефективних рекомендаційних механізмів, унаслідок чого кандидати отримують нерелевантні вакансії, а роботодавці — невідповідні відгуки. Це значно збільшує час підбору персоналу та знижує ефективність рекрутингу.

Сукупність зазначених причин призводить до низки негативних наслідків: зростання часу пошуку роботи для студентів і джуніор-фахівців, втрати мотивації користувачів, низької якості підбору кадрів, перевантаження HR-відділів та загального зниження ефективності ринку праці для початківців.

Дерево цілей є логічним продовженням аналізу та формується шляхом трансформації кожної проблеми у позитивну, досяжну ціль. Воно відображає, яким чином усунення першопричин призведе до досягнення бажаного стану системи. Структуру дерева цілей проєкту наведено на рис.

1.4

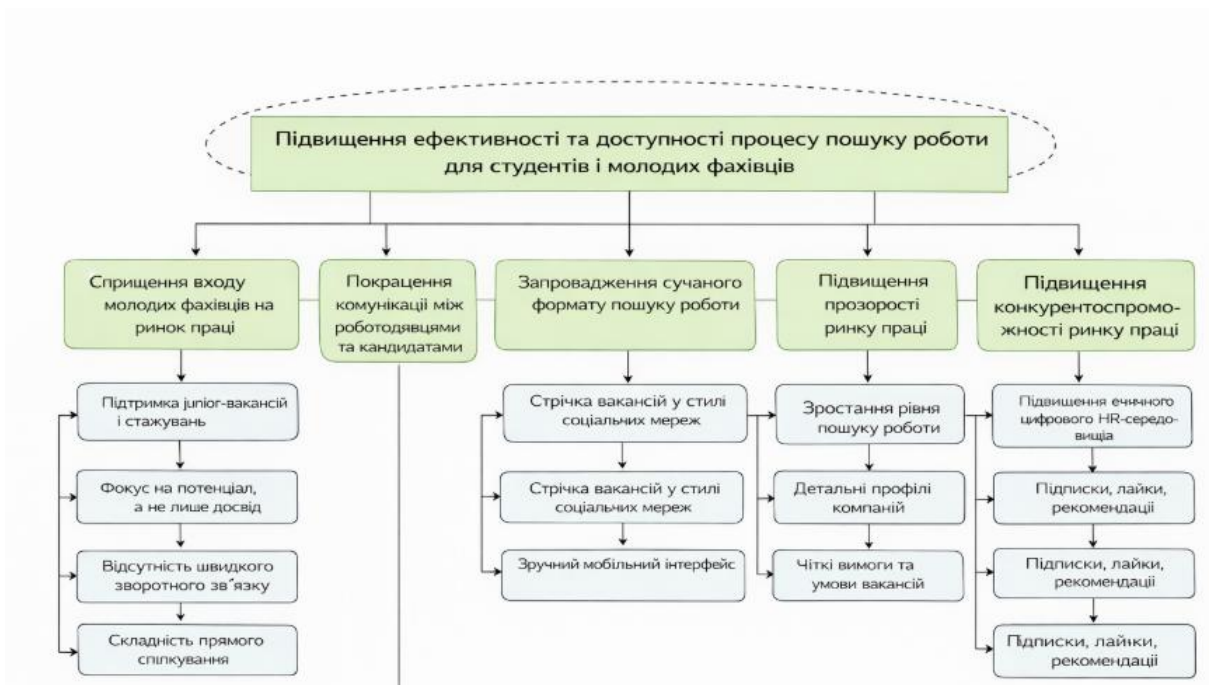


Рис 1.4 Дерево цїлей

Генеральною метою проєкту визначено створення сучасної веб-платформи для ефективної, зручної та персоналізованої взаємодії між студентами, молодими фахівцями та роботодавцями.

Досягнення цієї мети базується на реалізації кількох стратегічних напрямів.

Першим напрямом є централізація та структуризація даних, що передбачає створення єдиного цифрового середовища з профілями користувачів, вакансіями та історією взаємодії. Це забезпечує прозорість процесу пошуку роботи та спрощує аналіз кандидатів для роботодавців.

Другий напрям спрямований на підвищення зручності та залученості користувачів шляхом реалізації інтерфейсу у форматі соціальної мережі зі стрічкою вакансій, швидкими відгуками, чатами та візуально орієнтованою подачею інформації.

Третій напрям полягає у впровадженні персоналізованих рекомендацій і фільтрації, що дозволяє підвищити релевантність вакансій і відгуків, скоротити час пошуку роботи та оптимізувати рекрутингові процеси.

Таким чином, побудова дерева проблем і дерева цїлей підтвердила актуальність створення веб-платформи Workgram та дозволила чїтко

визначити пріоритетні напрями її розробки. Виявлені причинно-наслідкові зв'язки забезпечують спрямованість проєкту на усунення першопричин неефективності ринку праці для молодих спеціалістів, що створює надійну основу для подальшого планування та реалізації проєкту.

1.3 Огляд аналогів

Український ринок платформ для пошуку роботи сформувався навколо кількох ключових гравців, серед яких Work.ua, Rabota.ua, Jooble, LinkedIn, Djinni та низка менших нішевих сервісів. Більшість із них орієнтовані на масовий ринок, а їхні функціональні можливості зосереджені переважно на звичних механізмах: пошук вакансій, подача резюме, фільтри та формальні автоматизовані рекомендації.

Разом із тим, дедалі більшого значення набуває сегмент молодіжної аудиторії — студентів, випускників, стажерів, які роблять перші кроки у професійному розвитку. Саме ця категорія користувачів потребує не лише доступу до вакансій, а й підтримки, адаптації, простих та інтуїтивних інтерфейсів, можливості створення візуальних профілів, наявності навчальних матеріалів, системи рекомендацій та функціоналу взаємодії в форматі, який нагадує сучасні соціальні мережі.

У цьому контексті аналіз існуючих платформ дозволяє виявити їхню ефективність у задоволенні потреб молоді, а також окреслити прогалини, які можуть бути заповнені за рахунок спеціалізованих рішень. У таблиці нижче(див. *Таблиця 1.1*) наведено ключові характеристики платформ за критеріями:

Таблиця 1.1

Порівняння платформ

	Орієнтація	Наявність мобільного додатку	Наявність фільтра стажувань	Рекомендації	Оцінка користувачів (Play Market)

Work.ua	Масовий сегмент	+	-	-	4.6/5
Rabota.ua	Масовий сегмент	+	-	-	4.5/5
Jooble	Агрегатор	+	-	-	4.2/5
LinkedIn	Професій на мережа	+	+	+	4.3/5
Djinni.co	ІТ-сфера	+	-	+	4.7/5
Indeed	Масовий сегмент	+	-	+	4.6/5
Glassdoor	Масовий сегмент	+	-	+	4.4/5

Аналіз платформ:

- Work.ua та Rabota.ua є беззаперечними лідерами українського ринку працевлаштування, що підтверджується великою кількістю вакансій, роботодавців та активних резюме.

Статистичні дані (2024):

Work.ua: понад 2 млн активних резюме, близько 90 тис. роботодавців.

Rabota.ua: понад 1,5 млн резюме, приблизно 80 тис. роботодавців[12].

Попри обсяг та впізнаваність, обидві платформи мають спільні проблеми:

- недостатня орієнтація на молодіжний сегмент;
- відсутність спеціальних фільтрів для стажувань, практик або вакансій без досвіду;
- слаборозвинуті рекомендаційні алгоритми;
- обмежений візуальний інтерфейс, не схожий на формати соціальних мереж;

- формальний підхід до резюме, орієнтований переважно на ключові слова.

Таким чином, для студентів і випускників ці платформи можуть бути корисними, однак вони не закривають потреб у підтримці кар'єрного старту.

- Jooble — це міжнародний агрегатор, який збирає вакансії з різних джерел, включно зі світовими сервісами.

Особливості:

- відсутність власної розвиненої екосистеми;
- фокус на кількості, а не якості вакансій;
- мінімальна адаптація під окремого користувача;
- неможливість побудувати персональну траєкторію розвитку.

Jooble не пропонує студентам і молодим фахівцям спеціалізованих сервісів. Його головна перевага — масштаб і широта охоплення, проте недоліки полягають у низькій персоналізації..

- LinkedIn має розвинуту екосистему для B2B-спілкування, просування бренду[14], рекомендацій і нетворкінгу включаючи близько 3,5 млн українських профілів (за даними LinkedIn Insights), проте складний у використанні для студентів через розвинену інфраструктуру та відсутність локалізованих інструкцій.

Переваги:

- технологічно розвинений інтерфейс;
- рекомендаційна система; можливість формування бренду фахівця;
- розгалужена система нетворкінгу;
- велика кількість освітніх матеріалів.

Недоліки для молоді:

- складність навігації; висока конкуренція з боку досвідчених спеціалістів;
- відсутність локалізованих інструкцій для студентів;
- формальність комунікації;

- недостатній фокус на стажуваннях та практиках у локальному сегменті.

LinkedIn є потужним інструментом, але менш оптимальним для молодих користувачів, які не мають досвіду побудови професійного бренду.

- Djinni.co — приклад вузькоспеціалізованого рішення для IT-сфери. Має зручну систему анонімних профілів, але обмежений за галузями.
- Indeed і Glassdoor мають сильні позиції у світі, але в Україні часто дублюють інформацію з інших сайтів, мають низький рівень локалізації.

Недоліки:

- низька локалізація;
- дублювання вакансій із інших сайтів;
- нерелевантність пропозицій для початківців;
- слабка адаптація до українських нормативно-правових реалій.

Glassdoor вирізняється системою відгуків про роботодавців, але в українському сегменті цей функціонал використовується обмежено.

Статистика використання (Україна, 2024):

- Work.ua — понад 2 млн активних резюме, 90 тис. роботодавців[12]
- Rabota.ua — понад 1,5 млн резюме, 80 тис. роботодавців[13]
- LinkedIn — ~3,5 млн українських профілів[14] (дані LinkedIn Insights)
- Jooble — 1,2 млн щомісячних відвідувань (за SimilarWeb)

Більшість існуючих платформ не орієнтовані на молодь, не мають механізмів рекомендацій, перевірки профілів і не забезпечують належної підтримки перших кроків у професійному житті. Таким чином, є обґрунтована потреба у створенні спеціалізованої платформи, яка закrije цей сегмент.

1.4 Характеристика цільового користувача

Цільовими користувачами проекрованої веб-платформи є:

1. Студенти вищих навчальних закладів — особи, що перебувають на етапі здобуття освіти і зацікавлені в пошуку стажування, проходженні практики або першого місця роботи за спеціальністю.

2. Молоді фахівці — випускники ЗВО або фахових коледжів, які мають до 2 років досвіду роботи і шукають можливості для професійного розвитку, змін у кар'єрі або старту власного шляху.

3. Компанії та організації — юридичні особи, які мають потребу в нових кадрах, шукають молодих, мотивованих і навчальних працівників, готові інвестувати в стажування, внутрішнє навчання або менторство.

4. HR-фахівці та рекрутери — користувачі, які зацікавлені у зручному, структурованому та ефективному інструменті для відбору кандидатів, створення вакансій і ведення комунікації з пошукачами.

Усі вищезгадані групи потребують зручного, інтуїтивно зрозумілого інтерфейсу, інструментів фільтрації та комунікації, а також можливості формувати індивідуальні кар'єрні профілі та отримувати релевантні пропозиції в автоматичному режимі.

1.5 Постановка задачі, формулювання ТЗ, паспорт проєкту

На основі виявлених проблем та потреб цільової аудиторії, основним завданням проєкту є створення веб-платформи, яка дозволяє ефективно взаємодіяти роботодавцям і здобувачам на ранніх етапах професійного становлення.

Основні тренди:

1. Гейміфікація процесу пошуку роботи — рейтинги, бейджі, прогрес-бар.

2. AI-підбір вакансій — автоматична оцінка резюме та рекомендації на основі навичок.

3. Мікронавчання всередині платформи — короткі курси, відео, тести.

4. Побудова особистого бренду кандидата — профіль як «соціальна сторінка».

5. Інтеграція з соціальними мережами.
6. Використання мобільних застосунків як основного каналу.

Постановка задачі: Розробити веб-платформу з інтуїтивним інтерфейсом, системою фільтрації, рекомендаційним модулем і інструментами комунікації, орієнтовану на студентів, молодих фахівців і компанії-роботодавців[3].

Формулювання технічного завдання (ТЗ):

- Реєстрація користувачів двох типів: здобувачі та роботодавці;
- Можливість створення профілю та резюме з автозаповненням;
- Розміщення та пошук вакансій/стажувань;
- Вбудований чат для комунікації;
- Система верифікації роботодавців;
- Сторінка адміністратора з можливістю модерування;
- Мобільна адаптація інтерфейсу;
- Рекомендаційний алгоритм для персоналізованого підбору вакансій.

Паспорт проєкту:

- Назва проєкту: Workgram
- Замовник

Умовно — Центр кар'єри університету / стартап-команда

- Ключові цілі

Підвищення рівня працевлаштування молоді, спрощення пошуку кадрів, покращення прозорості відбору

- Стейкхолдери

Студенти, випускники, рекрутери, HR, кар'єрні центри

- Очікуваний результат

Пілотна версія веб-платформи з функціональним MVP

- Період реалізації: 3–4 місяці
- Критерії успіху

Стабільна робота, позитивний зворотний зв'язок, готовність до масштабування

РОЗДІЛ 2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА МОДЕЛЮВАННЯ СИСТЕМИ

2.1 Побудова концептуальної моделі системи

Розробка будь-якого програмного забезпечення, зокрема веб-платформи, починається з побудови концептуальної моделі, яка дозволяє узагальнено представити структуру системи, її основні компоненти, взаємозв'язки між ними та взаємодію із зовнішнім середовищем. Концептуальна модель є ключовим етапом проєктування, оскільки формує цілісне бачення майбутнього програмного продукту ще до початку його технічної реалізації та слугує основою для подальшого розроблення архітектури, бази даних і користувацьких інтерфейсів.

Розроблену концептуальну модель веб-платформи з пошуку роботи, яка функціонально та візуально орієнтована на принципи соціальної мережі типу Instagram, зображено на рис. 2.1. Платформа призначена для ефективної взаємодії між пошукачами роботи та роботодавцями шляхом публікації вакансій у форматі постів, формування персоналізованої стрічки, підписки на профілі, коментування, обміну повідомленнями та подання відгуків на вакансії.

У процесі розробки концептуальної моделі було виконано системний аналіз платформи, визначено її структуру, надсистему, підсистеми, зовнішні зв'язки та ключові функціональні компоненти.

1. Система

Веб-платформа для пошуку роботи у форматі соціальної мережі.

2. Аналіз системи з фізичної точки зору

Надсистема:

- пошукачі роботи;
- роботодавці та компанії;
- зовнішні сервіси (хмарна інфраструктура, платіжні системи, сервіси сповіщень).

Підсистеми:

- Веб-платформа – основна програмна підсистема, що забезпечує доступ до функціоналу публікації вакансій, перегляду стрічки, взаємодії між користувачами та обміну повідомленнями;
- Пошукачі роботи – користувачі, які створюють профілі, підписуються на роботодавців, переглядають вакансії та подають відгуки;
- Роботодавці – користувачі, які створюють корпоративні профілі, публікують вакансії, переглядають відгуки та комунікують із кандидатами;
- Адміністративна панель – спеціалізована підсистема для адміністраторів, що забезпечує контроль користувачів, контенту та технічного стану платформи.

3. Визначення зовнішніх зв'язків системи з надсистемою

Зовнішні фактори впливу:

- Пошукачі роботи: переглядають стрічку вакансій, коментують пости, подають заявки на вакансії, спілкуються з роботодавцями через чат;
- Роботодавці: публікують вакансії, керують профілем компанії, аналізують відгуки кандидатів, ведуть переписку;
- Зовнішні сервіси: забезпечують зберігання даних, надсилання push-сповіщень, обробку файлів резюме та стабільну роботу платформи.

4. Перелік підсистем

Веб-платформа:

- інтерфейс користувача (UI);
- система персоналізованої стрічки вакансій;
- модуль публікації вакансій (пости);
- система підписки на профілі;
- коментарі та взаємодія з постами;
- модуль миттєвих повідомлень (чат);
- система подання відгуків на вакансії (Apply).

Акаунт пошукача роботи:

- реєстрація та авторизація;

- створення та редагування профілю;
- перегляд стрічки вакансій;
- подання заявок із резюме;
- управління підписками;
- налаштування облікового запису.

Акаунт роботодавця:

- створення корпоративного профілю;
- публікація та редагування вакансій;
- перегляд і обробка відгуків;
- комунікація з кандидатами;
- аналітика активності вакансій.

Адміністративна панель:

- управління користувачами;
- модерація вакансій та коментарів;
- контроль дотримання правил платформи;
- моніторинг технічного стану системи;
- управління глобальними налаштуваннями.

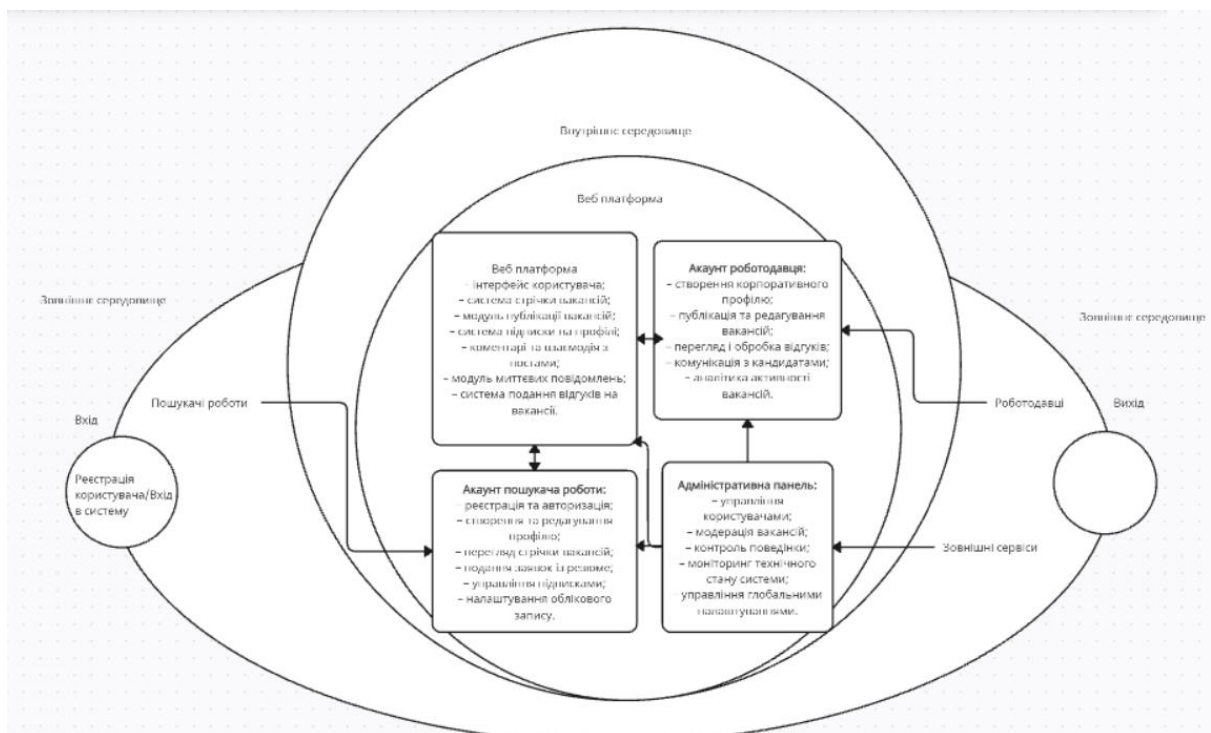


Рис 2.1 Концептуальна модель

2.2 Формалізація завдань пошуку та підбору кандидатів

Завдання пошуку та підбору кандидатів у цифровому середовищі є складним процесом, що поєднує елементи інформаційного пошуку, машинного навчання, теорії прийняття рішень та математичного моделювання. На сучасному ринку праці, де кількість резюме та вакансій може обчислюватися десятками чи сотнями тисяч, ручний аналіз та сортування даних стає неможливим, особливо для компаній, які ведуть активний найм або працюють з великою кількістю стажерів і молодих фахівців. Саме тому виникає потреба у формалізації завдань пошуку та підбору кандидатів, що дозволяє стандартизувати логіку роботи рекрутингової системи та автоматизувати ключові операції.

Формалізація полягає у перетворенні інформаційних та бізнес-процесів на математично описані операції та алгоритми, що дають змогу однозначно інтерпретувати вхідні дані (вимоги вакансії, характеристики кандидатів), перетворювати їх за визначеними правилами та отримувати структурований вихідний результат (список релевантних кандидатів, рейтинг відповідності, рекомендації тощо).

Основною метою формалізації етапу пошуку є зіставлення профілю здобувача з вимогами вакансії та обчислення рейтингу відповідності у кількісному вигляді. Це забезпечує прозорість, об'єктивність і відтворюваність процесів відбору, а також формує основу для подальшого впровадження систем рекомендацій (recommender systems).

У веб-платформі для працевлаштування основні задачі пошуку можуть бути згруповані в декілька функціональних категорій.

- Фільтрація кандидатів

Фільтрація є першим і базовим етапом пошуку. Вона передбачає відсіювання тих кандидатів, які не відповідають критично важливим вимогам вакансії:

- освіта (загальна або спеціалізована);

- рівень володіння іноземними мовами;
- досвід роботи;
- спеціалізація; технічні та професійні навички;
- локація або можливість релокації;
- тип зайнятості (повна, часткова, віддалена).

Фільтрація реалізується за допомогою набору логічних умов (предикатів), що мають чітку математичну інтерпретацію: кандидат включається в результати \leftrightarrow він задовольняє всі обов'язкові критерії.

- Пошук по ключовим значенням(за ключовими словами)

Пошук за ключовими словами актуальний у випадках, коли рекрутер не хоче будувати складні фільтри або коли потрібно знайти резюме за специфічними технологіями чи навичками, що можуть згадуватись у будь-якому місці профілю.

Приклади ключових слів:

- “Python”, “React”, “Node.js” — для IT-сфери;
- “Excel”, “SAP”, “CRM” — для економічних та офісних спеціальностей;
- “sales”, “communication”, “lead generation” — для вакансій менеджерського типу.

Результати пошуку зазвичай ранжуються за частотою входження ключових слів, їхньою позицією у резюме, контекстом та іншими параметрами, описаними у методах обчислення релевантності.

- Ранжування результатів

Після того як сформовано список кандидатів, необхідно відсортувати їх у порядку зменшення релевантності. Ранжування виконується за допомогою функції відповідності, що може враховувати:

- збіг навичок та технологій;
- відповідність досвіду роботи опису вакансії;
- релевантність освіти;
- історію успішних наймів;
- активність кандидата;

- повноту профілю;
- успішність співбесід у минулому (якщо доступно).

Таким чином, системи ранжування — це ядро алгоритмів автоматизованого рекрутингу.

- Рекомендаційні механізми

Рекомендаційні системи у рекрутингу працюють за принципами:

- content-based filtering — рекомендації на основі подібності профілю кандидата до профілів, які компанія шукала раніше;
- collaborative filtering — рекомендації на основі поведінки інших рекрутерів або кандидатів;
- hybrid models — поєднання двох методів.

Система може пропонувати:

- кандидатів для вакансії,
- вакансії для кандидатів,
- релевантних роботодавців,
- курси та матеріали для розвитку навичок.

У рамках платформи рекомендаційний модуль є важливою конкурентною перевагою.

- Аналіз сумісності (Matching Analysis)

Комплексний процес обчислення відповідності кандидата вимогам вакансії за низкою параметрів. Результатом є числова оцінка, зазвичай у діапазоні 0–100 %, де 100 % означає повну відповідність.

Параметри оцінювання можуть включати:

- технічні навички (hard skills);
- особистісні характеристики (soft skills);
- мотивацію кандидата;
- досвід роботи; освітній рівень;
- відповідність очікуваної зарплати;
- тип зайнятості;
- віддаленість локації.

Розширені системи також враховують:

- поведінкову історію кандидата;
- динаміку розвитку профілю;
- рекомендації попередніх роботодавців.

Щоб реалізувати ці завдання, необхідно формалізувати критерії пошуку. Наприклад:

Таблиця 2.1

Приклад критеріїв пошуку

Критерій	Тип	Приклад значення
Посада	Строковий тип	«Менеджер з продажу»
Досвід роботи	Цілочисельний тип	≥ 2 роки
Освіта	Строковий тип	Вища, IT-сфера
Навички	Масив строк	[HTML, CSS, JavaScript]
Локація	Строковий тип	«Київ»

Ці параметри використовуються для формування SQL-запитів або запитів до індексованого пошуку в Elasticsearch чи інших пошукових рушіях.

Формалізація процесів пошуку та підбору кандидатів є критичною складовою створення ефективної веб-платформи для ринку праці. Вона

дозволяє структурувати інформацію, побудувати логічні алгоритми пошуку та розробити інструменти автоматизованого рекрутингу. Завдяки цьому платформа не лише спрощує взаємодію між роботодавцями й кандидатами, але й значно підвищує швидкість та точність процесу підбору персоналу.

2.3 Математичне моделювання фільтрації вакансій і кандидатів

У сучасних веб-платформах для працевлаштування процес пошуку та підбору персоналу перетворився на складну структуровану задачу, яка об'єднує математичні методи, статистику, теорію інформаційного пошуку та алгоритми штучного інтелекту. Математичне моделювання відіграє ключову роль у побудові ефективних систем аналізу профілів кандидатів та вимог вакансій. Воно дозволяє формалізувати всі елементи процесу добору, перетворити неструктуровану інформацію у формалізовані вектори ознак, здійснити оптимізаційний відбір і провести ранжування результатів у кількісному, об'єктивному вигляді.

Математична модель необхідна тому, що традиційні підходи до рекрутингу (перегляд резюме вручну, суб'єктивні оцінки, довготривала обробка інформації) перестають бути ефективними в умовах, коли кількість резюме обчислюється сотнями тисяч, а ринок характеризується високою конкуренцією.

Математична модель дозволяє створити універсальний алгоритм, який забезпечить:

- об'єктивність підбору;
- автоматизацію зіставлення профілів і вакансій;
- формалізацію критеріїв оцінки;
- можливість впровадження інтелектуальних рекомендаційних механізмів.

Математичне моделювання фільтрації вакансій і кандидатів є ключовим елементом функціональної частини веб-платформи для майбутніх робітників і роботодавців. Завдяки цьому процесу забезпечується автоматизований, об'єктивний та ефективний відбір відповідних вакансій

для користувачів і, навпаки, релевантних кандидатів для роботодавців. Така фільтрація базується на наборі вхідних параметрів і правил, які можуть бути формалізовані за допомогою математичних моделей.

Основні математичні передумови моделі

Моделювання фільтрації вакансій і кандидатів базується на таких концепціях:

1. Теорія множин — для опису сукупностей кандидатів та вакансій.
2. Векторні простори — для опису характеристик (skills, experience, education).
3. Метрики подібності — для визначення рівня відповідності між профілем і вакансією.
4. Оптимізаційні задачі — для знаходження найрелевантніших кандидатур.
5. Моделі машинного навчання — для прогнозування успішності найму та побудови рекомендаційних механізмів.

Таким чином, процес фільтрації перетворюється на задачу пошуку максимуму функції відповідності у векторному просторі.

У найзагальнішому вигляді процес фільтрації можна подати як задачу оптимального зіставлення множини $V=\{v_1,v_2,\dots,v_n\}$ вакансій та множини $C=\{c_1,c_2,\dots,c_m\}$ кандидатів. Кожен елемент цих множин описується вектором характеристик: вакансія v_i має вектор $x_i=(x_{i1},x_{i2},\dots,x_{ik})$, кандидат c_j має вектор $y_j=(y_{j1},y_{j2},\dots,y_{jk})$, де k — кількість параметрів (наприклад, досвід роботи, освіта, бажана зарплата, тип зайнятості тощо).

Для оцінювання відповідності між вакансією та кандидатом застосовується функція релевантності або функція подібності $S(x_i,y_j)$, яка може бути реалізована як метрика відстані (наприклад, косинусна подібність або евклідова відстань) або через методи машинного навчання. Таким чином, формалізується задача фільтрації: $\max_{i,j} S(x_i,y_j)$ або ж у вигляді ранжування:

$R_j = \text{sort}_i(S(\mathbf{x}_i, \mathbf{y}_j))$, $R_j =$ рейтинг вакансій для кандидата c_j

$R_i = \text{sort}_j(S(\mathbf{x}_i, \mathbf{y}_j))$, $R_i =$ рейтинг кандидатів для вакансії v_i

Рис. 2.3.1 Формули вигляду ранжування

Для підвищення точності фільтрації можуть застосовуватись вагові коефіцієнти параметрів, які визначають пріоритетність окремих характеристик. Наприклад, ваговий вектор $w=(w_1, w_2, \dots, w_k)$ дозволяє адаптувати модель до потреб конкретного роботодавця або кандидата:

$$S(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^k w_i \cdot f(x_i, y_i)$$

де $f(x_i, y_i)$ — функція відповідності окремого параметра.

Рис. 2.3.2 Формули моделі до потреб конкретного роботодавця або кандидата

У разі використання алгоритмів машинного навчання (наприклад, класифікаторів чи нейронних мереж), фільтрація реалізується як задача прогнозування ймовірності того, що кандидат підходить для вакансії (або навпаки), на основі навченої моделі.

У системі можуть бути реалізовані додаткові обмеження у вигляді логічних умов (фільтрів), наприклад:

- географічне розташування (кандидати лише з певного регіону),
- рівень володіння мовами,
- наявність сертифікатів чи портфоліо тощо.

Застосування машинного навчання

Окрім класичних алгоритмів, у рекрутингу використовуються ML-моделі: 1. Логістична регресія

Прогнозує ймовірність успішності співбесіди.

2. Random Forest Оцінює значення різних характеристик.

3. Нейронні мережі

Можуть знаходити приховані залежності між навичками та успішністю.

1. Word2Vec / BERT

Створюють векторні представлення резюме та вакансій.

2.4 Основи використання рейтингових алгоритмів

У рамках розробки веб-платформи для майбутніх робітників та роботодавців одним із важливих інструментів підвищення якості взаємодії між сторонами є рейтингові алгоритми. Вони дозволяють автоматизовано оцінювати, впорядковувати та персоналізувати інформацію про користувачів, вакансії, відгуки та інші елементи системи.

Рейтингові алгоритми - це механізми обчислення кількісної оцінки об'єктів (кандидатів, вакансій, роботодавців), що дозволяють автоматизувати сортування та покращити релевантність результатів. Без рейтингових моделей система показує випадкові або малоцінні результати. Якщо ж впровадити рейтинги, платформа може працювати як інтелектуальний асистент з підбору роботи та персоналу.

Мета впровадження рейтингових алгоритмів

Рейтинги забезпечують:

- персоналізацію результатів пошуку;
- пріоритизацію найякісніших профілів;
- формування репутації користувачів;
- зменшення обсягу ручної перевірки контенту;
- боротьбу зі спамом, фейковими вакансіями та неякісними

резюме.

Значення рейтингових алгоритмів для платформи

Застосування рейтингових механізмів забезпечує:

- покращення релевантності результатів пошуку кандидатів або вакансій;
- підвищення довіри до користувачів через рейтингові оцінки профілю;

- автоматизовану фільтрацію сумнівного або неактивного контенту;

- персоналізовані рекомендації вакансій для користувачів.

Приклади використання на платформі

1. Рейтинг кандидатів

Для роботодавців важливо бачити найбільш релевантних та активних пошукачів. Формула рейтингу кандидата може враховувати:

- повноту профілю (заповнене резюме, фото, досвід, навички);
- активність (частота входу, оновлення профілю);
- кількість позитивних відгуків від попередніх роботодавців;
- відповідність ключових слів вакансії та резюме.

2. Рейтинг вакансій

Пошукачі з більшою ймовірністю зацікавляться вакансіями з високим рейтингом. Він може формуватись за критеріями:

- популярність серед інших кандидатів (кількість переглядів, відгуків);
- повнота та чіткість опису вакансії;
- авторитетність роботодавця;
- відповідність навичкам пошукача.

3. Рейтинг роботодавців

Враховує:

- швидкість відповіді;
- історію наймів;
- відгуки кандидатів;
- прозорість інформації про компанію.

4. Алгоритм рекомендацій

Система може пропонувати вакансії або кандидатів на основі комбінованого рейтингу, що враховує:

- історію пошуку користувача;
- схожість з іншими профілями;
- популярність серед аналогічних користувачів.

Потенційні труднощі

- Маніпулювання рейтингом (накручування відгуків, фальшиві дії[2]).
- Упередженість алгоритму (наприклад, через обмеженість даних або переоцінку популярності).
- Навантаження на систему при складних обчисленнях у реальному часі.

Оптимізація рейтингу

Система може враховувати:

- штрафи за неактуальність профілю;
- бонуси за високу активність;
- корекцію ваг через машинне навчання;
- антиспам-фільтри (запобігання накруткам).

Ефективне ранжування кандидатів і вакансій є фундаментальною частиною будь-якої сучасної платформи працевлаштування. Воно дозволяє системі не тільки відображати результати пошуку, але й визначати, які елементи є найбільш релевантними для користувача, базуючись на складних математичних моделях, поведінкових даних і семантичному аналізі контенту.

У веб-платформі використовуються комбіновані рейтингові підходи, які можуть працювати незалежно або як частини гібридної рекомендаційної системи.

1. PageRank-подібні моделі

PageRank — класичний алгоритм Google для визначення “важливості” вебсторінок. У контексті платформи працевлаштування він може застосовуватись для оцінки профілів користувачів, вакансій або компаній.

Принцип роботи:

Профіль вважається тим важливішим, чим більше важливих користувачів з ним взаємодіють.

У рекрутингу це означає:

- чим більше профіль отримує переглядів,

- чим частіше рекрутери переходять до нього,
- чим більше на нього підписані інші користувачі,
- чим частіше він з'являється у пошуку, тим вище його ранг.

Приклад застосування:

- Визначення “топ-талантів” серед кандидатів.
- Визначення найавторитетніших компаній.
- Формування рангових списків для рекомендацій.

PageRank-подібні моделі особливо корисні у спільнотах, де є зв'язки між користувачами (підписки, перегляди, лайки, коментарі), що робить їх релевантними й для системи, подібної до Instagram-орієнтованої платформи працевлаштування.

2. TF-IDF — оцінка ключових слів

TF-IDF (Term Frequency – Inverse Document Frequency) — класичний метод обчислення важливості ключових слів у документах.

У нашому випадку це:

- резюме,
- описи вакансій,
- портфоліо,
- коментарі.

TF-IDF у рекрутингу

Цей алгоритм:

- виділяє найбільш релевантні терміни з резюме,
- визначає специфічні професійні навички,
- допомагає підвищити точність пошуку за ключовими словами.

Наприклад: якщо слово “Python” часто з'являється в резюме і достатньо рідко в інших профілях, то його вагомість буде високою → кандидат вважається релевантнішим для Python-вакансій.

3. Collaborative Filtering (спільна фільтрація)

CF — основа рекомендаційних систем Netflix, Amazon, YouTube.

У рекрутингу вона відіграє роль у прогнозуванні:

- які кандидати можуть бути цікавими конкретному роботодавцю,

- які вакансії зацікавлять конкретного користувача.

Принцип:

“Користувачі зі схожою поведінкою мають схожі інтереси.”

Система аналізує:

- які вакансії переглядав кандидат,
- на які подавався,
- які профілі переглядали рекрутери,
- які компанії взаємодіяли з кандидатами.

На основі цієї інформації платформа пропонує:

- вам може сподобатися...
- цей кандидат схожий на тих, кого ви уже наймали...
- користувачі, які переглядали вакансію А, також переглядали

вакансію В.

4. Gradient Boosting (GBM, XGBoost, LightGBM)

Gradient Boosting — одна з найточніших моделей машинного навчання.

У системі він може використовуватися для:

- оцінки ймовірності успішного найму,
- прогнозування відповідності кандидата вакансії,
- оцінки впливу окремих характеристик профілю.

Переваги GBM у рекрутингу:

- справляється зі змішаними типами даних,
- може враховувати дуже багато параметрів,
- забезпечує високу точність прогнозів.

Наприклад, модель може навчитися, що:

- кандидати з певною комбінацією навичок мають високий шанс пройти технічну співбесіду;

- вакансії з конкретними параметрами викликають великий інтерес у молоді;

- активні профілі мають більшу конверсію в найм.

Інтеграція рейтингових алгоритмів у веб-платформу значно підвищує її ефективність, зручність та привабливість. Це дозволяє не лише автоматизувати процес підбору персоналу та пошуку роботи, але й забезпечити прозорість та об'єктивність взаємодії між користувачами. У межах курсового проекту їх використання є доцільним кроком до створення сучасного й конкурентоспроможного ІТ-рішення.

РОЗДІЛ 3. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ

3.1 Складання функціональних і нефункціональних вимог

Складання функціональних і нефункціональних вимог є критично важливим етапом при проєктуванні веб-платформи для майбутніх робітників та роботодавців. Саме ці вимоги формують технічне завдання для розробників, визначають можливості системи, очікувану поведінку та обмеження, яким вона має відповідати. Вимоги дозволяють забезпечити повноцінне функціонування платформи, відповідність її цільовому призначенню, а також зручність і безпеку для кінцевих користувачів.

Функціональні вимоги

Функціональні вимоги описують, що повинна робити система, які функції мають бути реалізовані. Вони орієнтовані на дії користувача, логіку обробки даних та інтеракцію з інтерфейсом. Функціональні вимоги окреслюють набір можливостей платформи, доступних для різних категорій користувачів: здобувачів, роботодавців і адміністратора.

До основних функціональних вимог веб-платформи входять:

1. Реєстрація та авторизація користувачів:

Система повинна забезпечувати:

- реєстрацію з верифікацією email або номеру телефону;
- вибір типу профілю (здобувач/роботодавець);
- вхід у систему через логін/пароль;
- можливість відновлення паролю.

Додаткові вимоги:

- можливість авторизації через Google/LinkedIn;
- CAPTCHA для захисту від ботів.

2. Управління профілем користувача:

Користувач повинен мати можливість:

- створювати і редагувати профіль;
- завантажувати фото, портфоліо, документи;
- редагувати резюме, включаючи:

- освіту,
- досвід роботи,
- навички,
- сертифікати,
- посилання на соціальні мережі;
- переглядати статистику відвідувань профілю.

Для роботодавця:

- інформація про компанію;
- логотип та корпоративна айдентика;
- можливість керувати вакансіями.

3. Робота з вакансіями

Система повинна дозволяти:

- створювати вакансії;
- встановлювати вимоги (досвід, навички, зарплатний діапазон, локація);
- редагувати та архівувати вакансії;
- переглядати відгуки кандидатів;
- застосовувати фільтри для пошуку резюме.

Здобувач може:

- переглядати вакансії;
- фільтрувати за локацією, зарплатою, типом зайнятості;
- відгукуватися на вакансію;
- зберігати вакансії в "обрані";
- переглядати історію відгуків.

4. Фільтрація та пошук:

- Реалізація пошукової системи з фільтрами для швидкого знаходження відповідних вакансій або кандидатів.
 - Автоматичні рекомендації на основі профілю.
- ### 5. Система відгуків та рейтингів:
- Кандидати можуть подавати заявки на вакансії.

- Роботодавці можуть залишати фідбек про кандидатів (і навпаки).

6. Система комунікацій

Вимоги:

- прями чат-повідомлення між кандидатом і роботодавцем;
- відправлення вкладень (файлів);
- push-сповіщення про нові повідомлення;
- статуси (прочитано / відправлено / успішно доставлено).

7. Аналітика та статистика:

Для роботодавців - кількість переглядів вакансії, кількість відгуків.

Для кандидатів - статистика перегляду резюме.

В цілому система повинна:

- фіксувати всі основні події:
 - авторизації,
 - редагування профілю,
 - створення вакансій;
- формувати статистичні звіти;
- надавати рекомендації щодо підвищення видимості профілю.

8. Рекомендаційна система

Система повинна автоматично:

- пропонувати вакансії користувачу;
- рекомендувати роботодавцю релевантних кандидатів;
- будувати персоналізовану видачу на основі:
 - навичок,
 - активності,
 - історії пошуків,
 - поведінкових патернів інших користувачів.

Використовуються:

- методи Content-Based Filtering;
- Collaborative Filtering;
- алгоритми ранжування.

9. Адміністративна панель

Адміністратор повинен мати можливість:

- переглядати всі профілі;
- блокувати або відновлювати користувачів;
- модерувати вакансії та коментарі;
- переглядати аналітику використання системи;
- керувати правами доступу.

Нефункціональні вимоги

Нефункціональні вимоги описують властивості системи, які не пов'язані безпосередньо з її функціональністю, але є важливими для якості, продуктивності та експлуатації платформи.

1. Продуктивність:

Для забезпечення масштабованості й стабільності платформи система повинна відповідати таким характеристикам:

Час відповіді сервера

- не більше 200–300 мс для основних операцій;
- не більше 1 секунди для пошуку з фільтрами.

Пропускна здатність

- підтримка 1000+ одночасних користувачів (MVP);
- можливість вертикального та горизонтального масштабування.

Навантаження бази даних

- оптимізація за індексами;
- кешування популярних запитів;
- розмежування навантаження (шардінг у майбутніх версіях).

Доступність (uptime)

- мінімум 99.5% для початкової версії;
- у майбутньому — до 99.9%.

2. Масштабованість:

○ Можливість розширення функціоналу без зміни основної архітектури.

- Підтримка інтеграції з мобільними застосунками.

3. Безпека:

Платформа обробляє персональні дані користувачів, тому має відповідати стандартам безпеки:

Шифрування даних:

- пароль — через bcrypt або Argon2;
- HTTPS — обов'язково.

Захист від популярних атак

- SQL Injection — параметризовані запити;
- XSS — екранування параметрів;
- CSRF — токени безпеки;
- DDoS — rate-limiting.

Резервне копіювання

- щоденний backup бази;
- зберігання в окремому середовищі.

Верифікація роботодавців

- документи;
- підтвердження юридичного статусу.

Логи безпеки

- фіксація IP при підозрілих діях;
- сповіщення про злом або спроби входу.

4. Доступність:

○ Цілодобова робота (24/7), час безвідмовної роботи (uptime) не менше 99,9%.

- Адаптивний дизайн для мобільних пристроїв, планшетів, ПК.

5. Використовуваність:

○ Інтуїтивно зрозумілий інтерфейс, мінімальна кількість кроків для виконання дій.

○ Мова інтерфейсу: українська, англійська (з можливістю додати інші).

6. Модульність:

Структура системи має бути побудована за принципами модульності - для полегшення обслуговування та оновлення.

7. Вимоги до UX/UI

Дизайн системи повинен відповідати принципам:

1. Простоти та інтуїтивності

Користувач без досвіду роботи з платформами працевлаштування повинен швидко розібратися.

2. Мінімалізму та однорідності стилю

Використання фірмової палітри, однакових компонентів інтерфейсу, зрозумілих іконок.

3. Доступності

- достатній контраст;
- можливість збільшення тексту;
- коректне відображення для людей із порушеннями зору.

4. Mobile-first дизайн

Оскільки понад 65% молоді користується мобільними пристроями.

5. Швидкість взаємодії

Користувач повинен виконати ключові дії за 1–2 кліки:

- знайти вакансію,
- відгукнутися,
- написати повідомлення.

6. Соціальна складова

Платформа має інтерфейс, схожий на Instagram:

- стрічка контенту,
- підписки,
- взаємодії,
- лайки або реакції (опційно).

Таким чином, правильно сформульовані функціональні та нефункціональні вимоги є основою для подальшого проектування, реалізації та тестування веб-платформи, забезпечуючи її ефективне функціонування та задоволення потреб користувачів.

Управління проектом зі створення веб-платформи для працевлаштування молодих фахівців потребує вибору такої методології, яка забезпечить високу гнучкість, адаптивність до змін, швидку реакцію на потреби користувачів і чітку структурованість процесів. Проект має ІТ-спрямованість, передбачає багатостадійний розвиток продукту, створення MVP, прототипування, тестування, поступове додавання функціоналу, а також тісну взаємодію між командою й кінцевими користувачами.

Проект веб-платформи оптимально реалізувати за допомогою гібридної моделі “Waterfall + Agile”, де:

Waterfall застосовується на етапах:

- аналізу предметної області,
- формування технічного завдання,
- побудови архітектури,
- моделювання БД,
- планування ризиків.

Scrum / Agile застосовується для:

- розробки програмної частини,
- UX/UI дизайну,
- тестування,
- адаптації під зворотний зв'язок студентів і роботодавців,
- формування MVP,
- впровадження нових модулів.

Причини вибору гібридної методології

1. Проект має високу змінність вимог.

Зміни логіки користувачів (студентів, HR) потребують гнучкого підходу.

2. Необхідність отримувати проміжні результати.

Працюючі версії MVP дозволяють тестувати платформу на реальних користувачах.

3. Потрібна документація (академічна, проектна).

Частина Waterfall забезпечує формальність структури магістерської роботи.

4. Висока інноваційність продукту.

Рекомендаційні алгоритми, чат, UX-механіки — все це тестується ітеративно.

Таким чином, гібридна модель є найбільш збалансованим і практичним рішенням для даного проєкту, поєднуючи контрольованість класичних підходів та гнучкість Agile.

3.2 Структурна декомпозиція робіт та організаційна структура проєкту

У межах розробки веб-платформи для взаємодії студентів, молодих фахівців та роботодавців структурна декомпозиція робіт (англ. Work Breakdown Structure, WBS) формується на основі гнучкої методології управління проєктами. На відміну від класичних каскадних моделей, де WBS визначається один раз на початку проєкту, у гнучкому підході структуру робіт дозволено адаптувати протягом усього життєвого циклу. Це зумовлено динамічною природою ринку працевлаштування та можливим уточненням вимог користувачів на етапах створення MVP.

У проєкті Workgram WBS побудована за принципом ітеративної декомпозиції, де високорівневі елементи (епіки) логічно розбиваються на менші пакети робіт (user stories та задачі). Структура містить п'ять основних рівнів, що охоплюють повний цикл створення системи — від аналізу потреб до підтримки платформи у продакшені.

Етап ініціації та аналітики

Цей етап включає всі підготовчі заходи, необхідні для старту проєкту.

До ключових робіт належать:

- дослідження ринку онлайн-платформ пошуку роботи;
- аналіз конкурентів (Work.ua, Djinni, LinkedIn, Indeed);
- визначення цільової аудиторії — студенти, джуніор-фахівці та роботодавці;
- формування початкової візії продукту;

- складання початкового Backlog Product;
- оцінка технічних та ресурсних ризиків.

Отримані результати забезпечують основу для розробки наступних спринтів.

Етап проектування

На цьому етапі створюється технічний та візуальний фундамент платформи. Основні завдання:

- розробка UX/UI прототипів: стрічки вакансій, профілю користувача, форми створення вакансії, чату;
- моделювання логічної структури даних та формування ER-діаграми;
- визначення архітектури застосунку (клієнт–сервер, REST API, WebSocket-комунікація);
- вибір технологічного стеку (React, Node.js, PostgreSQL);
- проектування ролей і прав доступу: адмін, роботодавець, здобувач.

На цьому ж етапі проводиться побудова Use Case діаграм для ключових сценаріїв: реєстрація, створення вакансії, відгук, чат, модерація.

Етап розробки

Розробка платформи є найбільш ресурсомістким етапом, який декомпозиється на функціональні модулі. Основні пакети робіт:

- розробка модуля реєстрації та автентифікації;
- створення модуля профілів користувачів;
- реалізація стрічки вакансій (аналог Instagram feed);
- інтеграція системи підписок та рекомендацій;
- створення модуля створення вакансій (Create Job Post);
- розробка чату для роботодавців і пошукачів;
- реалізація адмін-панелі;
- інтеграція алгоритмів ранжування і фільтрації вакансій;
- адаптація інтерфейсу під мобільні пристрої.

Кожний модуль реалізовується у рамках окремих ітерацій (спринтів), що дозволяє поступово збільшувати функціональність MVP

Етап тестування

Тестування забезпечує якість і безпомилковість роботи платформи перед запуском. У межах цього етапу виконуються:

- модульне тестування бекенду та фронтенду;
- інтеграційні тести для перевірки взаємодії стрічки, чату, профілів;
- тестування продуктивності: швидкість завантаження стрічки, робота WebSocket;
- тестування безпеки: SQL-ін'єкції, XSS, перевірка паролів;
- UX-тестування інтерфейсів на групі користувачів.

Етап впровадження та підтримки

Фінальний етап включає виведення продукту у реальне середовище:

- розгортання сервера на хмарній платформі (AWS/Render/Google Cloud);
- запуск MVP версії;
- моніторинг логів, навантаження та поведінки користувачів;
- збирання зворотного зв'язку і формування нового Backlog;
- технічна підтримка користувачів та виправлення знайдених помилок.

WBS забезпечує прозору логіку виконання проєкту, дозволяє

керувати строками, ресурсами та якістю кінцевого продукту.



Рис 3.1 Структурна декомпозиція робіт

Організаційна структура проекту (OBS)

Організаційна структура проекту (англ. Organizational Breakdown Structure, OBS) визначає принципи розподілу відповідальності, ролі членів команди та канали комунікації у процесі створення веб-платформи Workgram.

Оскільки продукт розробляється у сфері HRTech — з високою конкуренцією, потребою в швидких релізах та регулярним оновленням функціоналу — для проекту обрано Agile/Scrum-підхід. Він забезпечує гнучкість, можливість швидкої зміни пріоритетів та оптимальний розподіл задач у спринтах.

Стратегічний рівень

Визначає бізнес-цілі, бюджет і ризики.

Project Manager (PM)

- планує строки та ресурси,
- контролює виконання бюджету,
- забезпечує взаємодію зі стейкхолдерами,
- звітує про прогрес.

Product Owner (PO)

- формує бачення продукту,
- складає та пріоритезує Backlog Product,
- контролює відповідність функціоналу очікуванням користувачів.

Тактичний рівень

Забезпечує організацію процесу розробки відповідно до Scrum.

Scrum Master

- проводить планування спринтів, daily-мітинги, ретроспективи;
- усуває перешкоди для команди;
- забезпечує дотримання методології Agile.

Виконавчий рівень (крос-функціональна команда)

Саме тут відбувається створення продукту.

UI/UX дизайнер

- проєктує інтерфейси стрічки вакансій, профілю користувача, чату;
- забезпечує доступність і зручність взаємодії.

Frontend Developer

- реалізує інтерфейс платформи на React;
- працює з API, WebSocket-чатом, Infinite Scroll.

Backend Developer

- створює серверну логіку на Node.js;
- реалізує модуль ранжування вакансій;
- забезпечує безпеку та роботу бази даних.

QA Engineer

- тестує веб-платформу,
- перевіряє функціональність, продуктивність, UX.

Спеціаліст з кібербезпеки

- відповідає за захист даних користувачів,
- проводить аудит безпеки.

Системний аналітик

- трансформує бізнес-вимоги у технічні специфікації,

- формує Use Case, User Stories, діаграми.

Консультативний рівень

Створює експертну підтримку проекту.

HR-консультант

- допомагає формувати логіку підбору кандидатів,
- перевіряє коректність алгоритмів ранжування.

Юридичний консультант

- готує політику конфіденційності, умови використання,
- контролює відповідність GDPR.

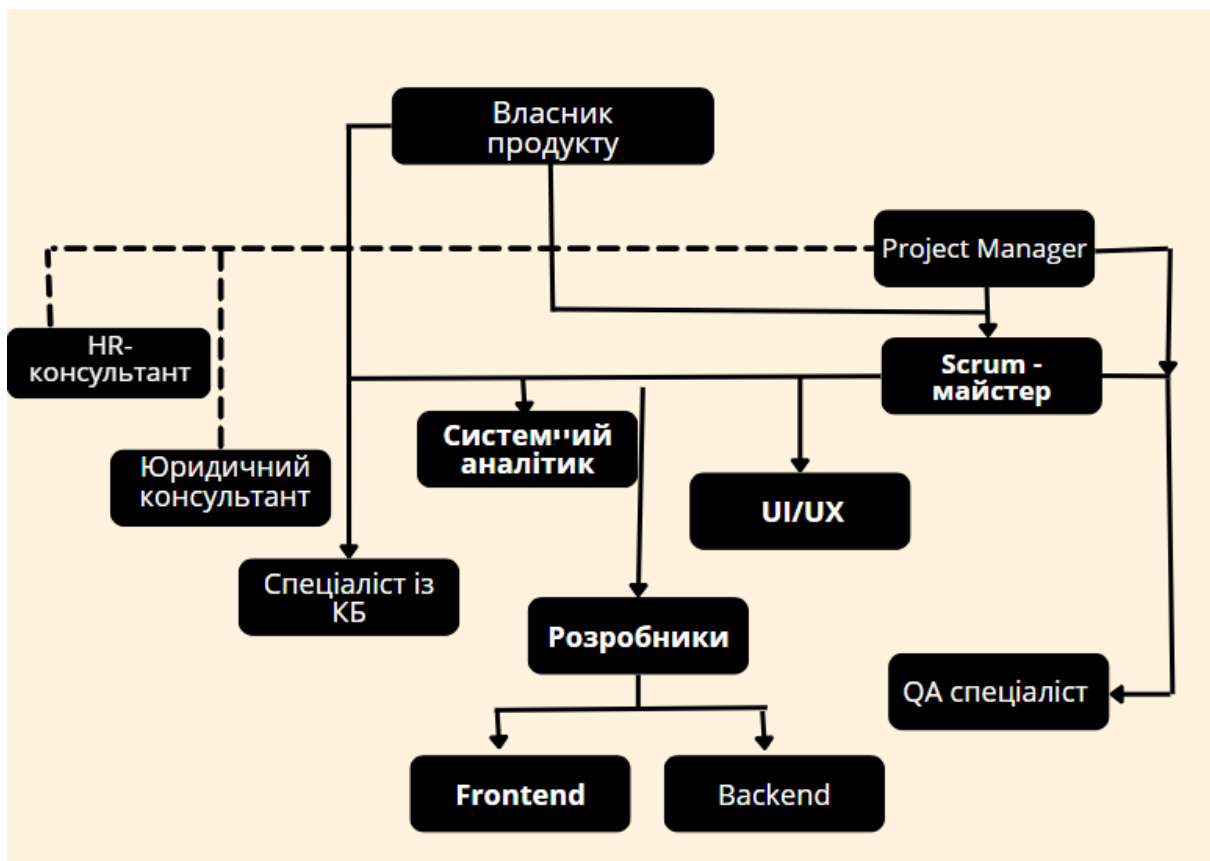


Рис 3.2 Структурна декомпозиція робіт

3.3 Use Case діаграми (реєстрація, пошук, чат, вакансії тощо)

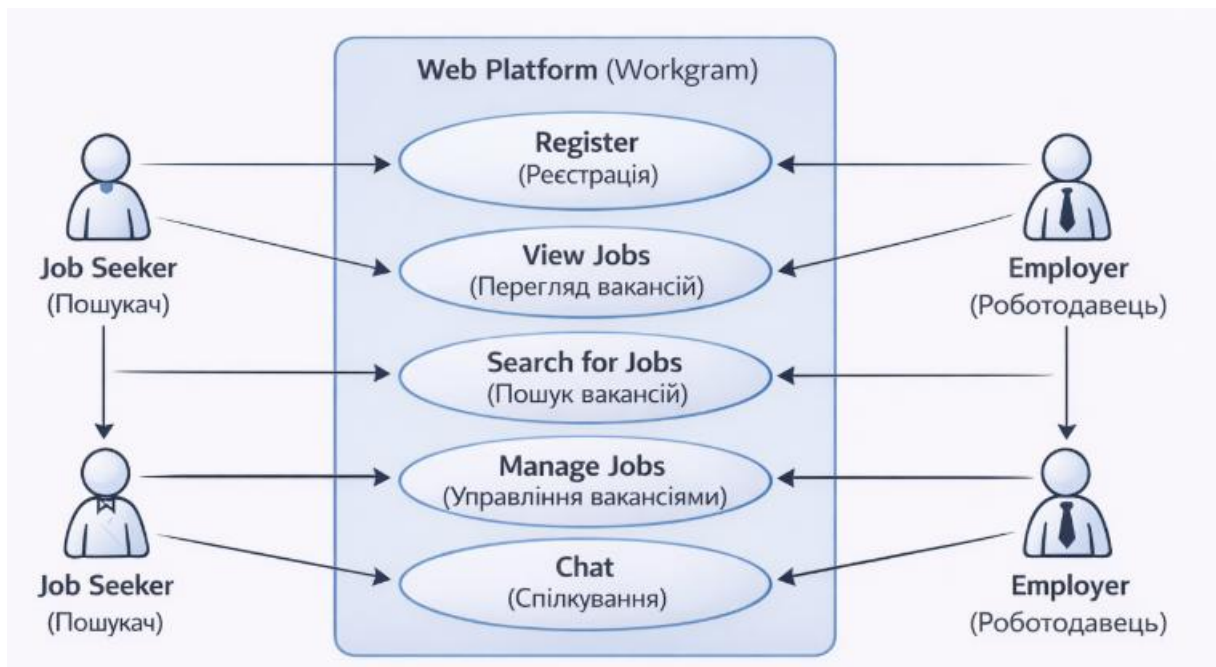


Рис 3.3 Use Case діаграма

Use Case діаграма, представлена вище, є ключовим інструментом для візуалізації основних функцій веб-платформи, яка розробляється в межах курсової роботи на тему «Розробка проєкту створення веб-платформи для майбутніх робітників та роботодавців». Вона демонструє взаємодію між основними користувачами системи — Job Seeker (Пошукач) та Employer (Роботодавець) — і тими функціональними можливостями, які надає платформа.

Основні :

1. Job Seeker (Пошукач):
 - користувач, який створює резюме, переглядає вакансії, подає заявки та спілкується з роботодавцями.
2. Employer (Роботодавець):
 - компанія або її представник, який публікує вакансії, переглядає кандидатів і веде з ними комунікацію.

Основні варіанти використання (Use Cases):

- Register (Реєстрація): Обидві категорії користувачів проходять процес реєстрації, створюючи відповідні профілі. Це перший крок для отримання доступу до сервісів платформи.

- Search for Jobs (Пошук вакансій):
Актуально переважно для пошукачів. Реалізується через функціонал пошуку із фільтрами (галузь, зарплата, тип зайнятості, місце розташування тощо).

- View Jobs (Перегляд вакансій):
Обидва актори можуть переглядати оголошення. Пошукачі — щоб знайти підходящу роботу, а роботодавці — щоб перевірити конкурентні оголошення.

- Manage Jobs (Управління вакансіями):
Роботодавець має змогу створювати, редагувати або видаляти вакансії, а також відстежувати активність за ними (відгуки, перегляди тощо).

- Chat (Спілкування):
Двостороння комунікація між пошукачем і роботодавцем, яка забезпечує швидкий обмін інформацією, уточнення деталей співбесіди чи умов праці.

Дана діаграма Use Case відіграє важливу роль у структуризації вимог до інформаційної системи. Вона:

- дозволяє чітко визначити ролі користувачів та їхні цілі;
- служить основою для подальшого проектування інтерфейсів і бізнес-логіки;
- дає змогу перевірити повноту функціональних вимог (відповідність задачам пошуку, взаємодії, створення вакансій);
- полегшує комунікацію між замовником, розробником і аналітиком у проєкті;
- слугує підґрунтям для створення технічної документації та тест-кейсів.

Таким чином, побудова Use Case діаграми є невіддільним етапом розробки веб-платформи, що сприяє системному підходу до аналізу функцій майбутньої системи та підвищує якість проектування.

3.4 Управління ризиками

Успішна реалізація веб-платформи « Workgram » — цифрового сервісу для взаємодії роботодавців, студентів та молодих фахівців - неможлива без системного та проактивного управління ризиками.

Особливість проєкту полягає у використанні хмарних технологій, алгоритмів рекомендацій, зберігання персональних даних користувачів та інтеграції функціоналу, подібного до соціальних мереж, що значно розширює можливий спектр загроз.

Процес управління ризиками базується на класичній моделі, що включає три основні етапи:

1. Ідентифікація ризиків,
2. Кількісна оцінка,
3. Розробка плану реагування.

У ході роботи було ідентифіковано 28 потенційних ризиків, класифікованих за сферами впливу:

Класифікація ризиків Workgram

1. Технічні
 - збої у роботі алгоритмів рекомендацій;
 - проблеми продуктивності стрічки вакансій;
 - помилки чату (WebSocket);
 - інтеграційні проблеми (соцмережі, email-верифікація).
2. Управлінські
 - недооцінка термінів спринтів;
 - нестача технічної документації;
 - невідповідність команди методології Agile.
3. Операційні та юридичні
 - порушення вимог GDPR і Закону України «Про захист персональних даних»;
 - витік даних;
 - відмова хмарного хостингу (AWS / Google Cloud).

4. Фінансові
 - перевищення бюджету хмарних сервісів;
 - змінність валютного курсу;
 - нестача коштів на просування MVP.
5. Взаємодія з користувачами та партнерами
 - негативні відгуки на ранню версію;
 - неактивність користувачів;
 - низька конверсія у відгуки.
6. Форс-мажорні
 - воєнна ситуація в Україні;
 - тривалі відключення електроенергії.

Таблиця 3.2

Ідентифікація ризиків

№	Тип ризику	Ризикова подія	Сила впливу	Керованість
1	Технічні	Низька продуктивність стрічки вакансій на слабких пристроях	Висока	Висока
2	Технічні	Помилки у WebSocket-чаті (затримки, дублікати повідомлень)	Середня	Висока
3	Технічні	Збої алгоритмів ранжування (неправильні рекомендації)	Висока	Середня
4	Технічні	Проблеми інтеграції email-верифікації	Середня	Середня
5	Управлінські	Нечітко визначені вимоги	Висока	Висока
6	Управлінські	Відсутність актуальної документації	Середня	Висока
7	Операційні	Збій хмарного хостингу	Висока	Низька
8	Операційні	Витік персональних даних	Критична	Середня
9	Юридичні	Порушення GDPR	Висока	Висока
10	Взаємодія	Негативні відгуки через баги MVP	Висока	Висока
11	Фінансові	Перевищення бюджету на хмарні ресурси	Висока	Низька
12	Форс-мажорні	Блекаути	Висока	Низька
13	Форс-мажорні	Воєнні ризики	Критична	Низька

Кількісне оцінювання ризиків

Оцінювання проводилося за 10-бальною шкалою за критеріями:

1. Затримка у часі
2. Фінансові збитки
3. Ймовірність прояву
4. Частота повторення

Комплексний показник визначався формулою:

$$\text{Важливість ризику} = \text{Фінансовий вплив} \times \text{Ймовірність} \quad (3.1)$$

Таблиця 3.3

Результати оцінювання ризиків Workgram

№	Ризикова подія	Затримки	Фінансові витрати	Ймовірність	Частота	Важливість
1	Низька продуктивність стрічки	7	7	6	5	42
2	Помилки WebSocket	5	5	6	5	30
3	Неправильні рекомендації	8	8	7	6	56
4	Нечіткі вимоги	8	6	7	5	42
5	Збій хмарного хостингу	9	9	2	1	18
6	Витік даних	10	10	5	3	50
7	Негативні відгуки MVP	8	6	7	8	42
8	Перевищення бюджету	9	9	6	4	54
9	Блекаути	8	6	9	8	54
10	Війна	10	10	9	9	90

План протидії ризикам

Розроблено план реагування, який включає:

- Профілактичні заходи
- Симптоми прояву ризику (тригери)

- Дії при появі симптомів
- Реактивні дії при реалізації ризику

Таблиця 3.4

Приклади протиризикових заходів Workgram

№	Ризик	Профілактика	Симптом	Дії при симптомі	Дії при проблемі
1	Низька продуктивність стрічки	Оптимізація запитів, кеш	Затримки >1 сек	Перезавантаження мікросервісу	Масштабування
2	Помилки рекомендацій	Тестування моделей ML	Нерелевантні вакансії	Перенавчання моделі	Тимчасове вимкнення алгоритму
3	Витік даних	Шифрування, аудит	Підозрілі запити	Блокування IP	Аварійний режим + повідомлення ДП
4	Перевищення бюджету	Моніторинг AWS/GC	80% бюджету за 50% роботи	Оптимізація інфраструктури	Тимчасове замороження проєкту
5	Блекаути	Використання Starlink/Eco Flow	Графіки відключень	Перехід на асинхронний графік	Перерозподіл задач

3.5 Бюджет проєкту

Економічне планування розробки веб-платформи «Workgram» здійснюється на основі моделі Time & Material (T&M), що дозволяє оплачувати фактично затрачений час команди та забезпечує гнучкість у ситуації змін вимог, появи нових функцій або коригування пріоритетів з боку стейкхолдерів.

T&M — оптимальна модель для створення цифрових продуктів класу MVP, оскільки вона дозволяє швидко змінювати функціональність на основі зворотного зв'язку користувачів, тестування гіпотез і результатів експериментів з алгоритмами рекомендацій.

Бюджет розраховано для життєвого циклу тривалістю 12 місяців, який включає етапи:

- ініціації та аналізу,

- активної розробки MVP,
- тестування та оптимізації,
- релізу,
- технічної підтримки та супроводу.

Основою фінансового планування є визначення складу команди, ступеня залученості кожного учасника та оцінка навантаження за ролями. Для оптимізації бюджету проєкту використовується гібридна модель залучення, де:

- технічна команда бере участь протягом усього періоду розробки з різним навантаженням,
- адміністративні ролі працюють частково,
- експерти (UI/UX, безпека, юридичні консультанти) залучаються короткостроково.

Таблиця 3.5

**План залучення та навантаження людських ресурсів у
Workgram**

Роль	Тип зайнятості	Навантаження	Тривалість участі	Примітка
Backend Developer	Змінна	100% → 40%	12 міс.	6 міс. – розробка API, сервісів рекомендацій, WebSocket-чату; 6 міс. – оптимізація та підтримка
Frontend Developer (web + адаптив)	Змінна	100% → 30%	12 міс.	6 міс. – основна розробка інтерфейсів; 6 міс. – оновлення та інтеграція нових модулів
QA Engineer	Часткова	50%	12 міс.	Автотести, регресія, тестування

				продуктивності стрічки та чату
Project Manager (PM)	Часткова	50%	12 міс.	Планування спринтів, контроль ризиків, координація
Product Owner (PO)	Часткова	20%	12 міс.	Формування беклогу, пріоритизація функцій, проведення демо
Scrum Master (SM)	Часткова	20%	12 міс.	Фасилітація командних подій, зняття блокерів
System Analyst	Проектна	100%	2 міс.	Аналіз ринку, формалізація вимог, специфікації
UI/UX Designer	Проектна	100%	2 міс.	Дизайн стрічки вакансій, профілів, чату, сторінки вакансії
Specialist in Cybersecurity	Проектна	—	1 міс.	Перевірка безпеки, аудит шифрування та авторизації
Маркетинговий консультант	Проектна	—	2 тижні	Проведення первинного маркетингового аудиту, запуск рекламних кампаній
Юридичний консультант	Проектна	—	2 тижні	Політика конфіденційності, GDPR, обробка даних користувачів

Розрахунок фонду оплати праці

Суми розраховані на основі ринкових ставок ІТ-фахівців в Україні.

Таблиця 3.6

Фонд оплати праці команди CyberStart

Роль	ЗП/міс (грн)	Формула	Всього (грн)
Backend Developer	70 000	$(70\,000 \times 6) + (28\,000 \times 6)$	588 000
Frontend Developer	45 000	$(45\,000 \times 6) + (13\,500 \times 6)$	351 000
QA Engineer	35 000	$35\,000 \times 0.5 \times 12$	210 000
Project Manager	45 000	$45\,000 \times 0.5 \times 12$	270 000
Product Owner	48 000	$48\,000 \times 0.2 \times 12$	115 200
Scrum Master	42 000	$42\,000 \times 0.2 \times 12$	100 800
System Analyst	40 000	$40\,000 \times 2$	80 000
UI/UX Designer	38 000	$38\,000 \times 2$	76 000
Cybersecurity Specialist	25 000	$25\,000 \times 1$	25 000
Маркетинговий консультант	22 000	$22\,000 \times 0.5$	11 000
Юрист	20 000	$20\,000 \times 0.5$	10 000

Сумарно фонд оплати праці: 1 837 000 грн

Операційні витрати

Включають витрати на хостинг, інструменти дизайну, DevOps-сервіси, аналітику та просування.

Таблиця 3.7

Операційні та маркетингові витрати CyberStart

Витрата	Опис	Вартість/рік (грн)
Хмарна інфраструктура (AWS/Google Cloud)	Сервери, база даних, CDN, S3, email-сервіси	180 000
Ліцензії ПЗ	Figma, Jira, Slack, GitHub Actions	50 000
Домен та SSL	cyberstart.ua + сертифікати безпеки	3 000
Аналітика (Mixpanel/GA4 pro)	повна інтеграція аналітичних сервісів	12 000
Маркетинг та реклама	запуск MVP, таргет, SMM, колаборації в TikTok	300 000
Бухгалтерський супровід	зовнішній аутсорсинг	55 000

Сумарно операційні витрати: 600 000 грн

Резервний фонд

Для стабільності проєкту закладено 10% від основних витрат як резерв.

Таблиця 3.8

Зведений бюджет проєкту CyberStart

Стаття витрат	Сума (грн)	Частка (%)
Фонд оплати праці	1 837 000	67%
Операційні витрати	600 000	22%
Основні витрати	2 437 000	89%
Резервний фонд (10%)	243 700	11%
ЗАГАЛЬНИЙ БЮДЖЕТ	2 680 700 грн	100%

3.6 Логічна модель (ER-діаграма)

Логічна модель бази даних – це детальне представлення структури даних, яке включає таблиці, їх атрибути, зв'язки між таблицями, первинні (PK) та зовнішні (FK) ключі. Вона розробляється незалежно від конкретної СКБД, забезпечуючи нормалізацію даних та усунення надлишковості[11].

Призначення логічної моделі:

- Визначає основні дані, необхідні для роботи веб-платформи.
- Створює фундамент для фізичного проєктування бази даних.
- Забезпечує коректну побудову зв'язків між учасниками платформи.
- Дає змогу реалізувати механізми пошуку, відгуків, перегляду профілів і спілкування.

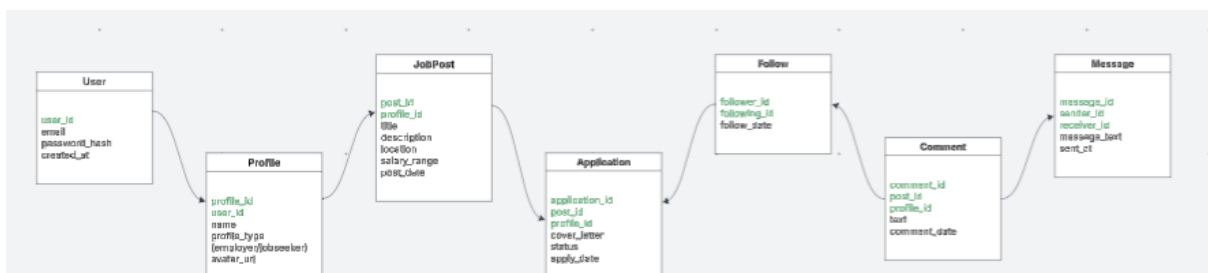


Рис. 3.4 Логічна (дatalogічна) модель бази даних

ER-модель має таку структуру:

- В центрі — «User» і «Profile»

- Від профілю відходять основні функціональні гілки:
 - створення вакансій → JobPost
 - відгуки → Application
 - взаємодія → Follow, Message, Rating, Comment
 - збережені вакансії → SavedJobs
 - файли → ResumeFile

Система побудована за модульним принципом, що дозволяє:

- масштабувати окремі елементи;
- розподіляти навантаження на різні сервери;
- ізолювати дані для мікросервісної архітектури у майбутньому.

Перед побудовою логічної моделі було проведено нормалізацію даних. Це забезпечує:

- мінімізацію дублювання інформації;
- підвищення цілісності даних;
- забезпечення узгодженості при оновленнях;
- оптимальність запитів;
- можливість масштабування.

Таблиці системи відповідають таким нормальним формам:

1 нормальна форма (1НФ):

- усі атрибути атомарні;
- відсутні повторювані групи;
- кожен запис має унікальний первинний ключ.

2 нормальна форма (2НФ):

- усі не ключові атрибути залежать від повного первинного ключа;
- таблиці не містять часткових залежностей.

3 нормальна форма (3НФ):

- атрибути залежать лише від первинного ключа;
- немає транзитивних залежностей (напр., таблиця «User» не зберігає дані профілю).

Переваги логічної моделі

1. Гнучкість структури

В окремі таблиці виділені:

- комунікації,
- соціальні зв'язки,
- рейтинг,
- файли.

Це робить модель майже необмежено масштабованою.

2. Оптимізація пошуку

Пошук прискорюється завдяки індексам на:

- email,
- skills,
- required_skills,
- title,
- location.

3. Сумісність із рекомендаційними алгоритмами

Введено окремі поля для:

- історії взаємодій
- навичок у форматі JSON
- рейтингу

4. Безпечність і розмежування доступів

Реалізовано поділ User/Profile.

5. Підтримка майбутньої аналітики

Сутність Log зберігає історію подій.

РОЗДІЛ 4. ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Архітектура веб-платформи

Веб-платформа побудована на основі клієнт-серверної архітектури, що передбачає розділення на фронтенд, бекенд і базу даних:

Клієнтська частина (Frontend):

- Реалізована з використанням HTML5, CSS3, JavaScript, React.
- Забезпечує інтерфейс користувача: реєстрація, створення резюме, перегляд вакансій, взаємодія з повідомленнями тощо.
- Спілкується з сервером через API-запити (REST або GraphQL).

Серверна частина (Backend)[11]:

- Побудована на Node.js (Express) або Python (Django).
- Виконує авторизацію, обробку запитів, формування відповідей, перевірку прав доступу.
- Реалізує логіку рейтингу, пошуку, рекомендацій тощо.

База даних:

- Використовується PostgreSQL.
- Таблиці: Users, Resumes, Vacancies, Applications, Messages, Feedbacks.
- Забезпечено нормалізацію БД, зовнішні ключі, індексація.

Допоміжні сервіси:

- Модуль email-розсилки;
- Система логування;
- Хмарне сховище для фото/резюме.

Інтерфейси

Основні інтерфейси користувача:

- Форма входу/реєстрації;
- Кабінет роботодавця (створення вакансій, перегляд відгуків);
- Кабінет шукача (створення резюме, перегляд вакансій, відгуки);
- Сторінка пошуку та фільтрації результатів;

- Інтерфейс для повідомлень.

4.2 Розробка бази даних проекту

Проектування бази даних є ключовим етапом створення веб-платформи, оскільки саме БД забезпечує структуроване збереження профілів користувачів, вакансій, відгуків, повідомлень, заявок на вакансії та всіх бізнес-процесів, які формують функціональність системи.

База даних платформи виконує такі основні функції:

- збереження профілів роботодавців та пошукачів;
- збереження вакансій (аналог постів);
- збереження підписок між користувачами (аналог Instagram follow);
- обробка відгуків на вакансії;
- збереження повідомлень у чатах;
- формування стрічки вакансій;
- забезпечення роботи рекомендаційних алгоритмів;
- логування діяльності користувачів.

Для зберігання даних обрана реляційна СКБД PostgreSQL, яка підтримує складні зв'язки, індексацію текстових полів, JSON-атрибути для гнучких структур та надійність ACID-транзакцій.

Концептуальна модель бази даних

Концептуальна модель — це високорівневе уявлення про структуру даних без урахування конкретної реалізації. Вона описує основні сутності та взаємозв'язки між ними.

У предметній області «веб-платформа для роботодавців і майбутніх робітників» ключовими сутностями є:

Основні сутності

1. User (Користувач)

Описує загальний профіль користувача (пошукач або роботодавець).

Містить основну інформацію: ім'я, email, роль, фото, біографія.

2. Vacancy (Вакансія / Публікація)

Аналог постів в Instagram.

Містить опис роботи, вимоги, зарплату, фото/банер, теги, дату створення.

3. Follow (Підписка)

Self-reference зв'язок User → User.

Дозволяє формувати персоналізовану стрічку вакансій.

4. Application (Відгук на вакансію)

Фіксує подачу заявки на вакансію.

Містить прикріплене резюме, статус розгляду.

5. Chat (Чат)

Описує діалог між двома користувачами.

6. Message (Повідомлення)

Повідомлення всередині чату (аналог Instagram Direct).

7. Comment (Коментар)

Коментар до вакансії-публікації.

8. Company (Компанія)

Додаткова сутність для роботодавців із корпоративними профілями.

9. Tag (Тег)

Теми вакансій (наприклад, “Python”, “Design”, “Remote”).

10. VacancyTag (зв'язок «багато-до-багатьох»)

Дозволяє прикріплювати кілька тегів до вакансії.

Зв'язки між сутностями

- User — Vacancy (1:N): один роботодавець може створювати багато вакансій.
- User — Follow — User (M:N): користувач може підписуватись на інших.
- Vacancy — Application — User (M:N): один користувач може податися на багато вакансій.
- Vacancy — Comment — User (1:N): коментарі до публікацій.

- User — Chat — User (M:N): чат між двома користувачами.
- Chat — Message (1:N): у чаті багато повідомлень.
- Vacancy — VacancyTag — Tag (M:N): система тегів для пошуку.
- User — Company (1:1): роботодавець може бути прив'язаний до компанії.

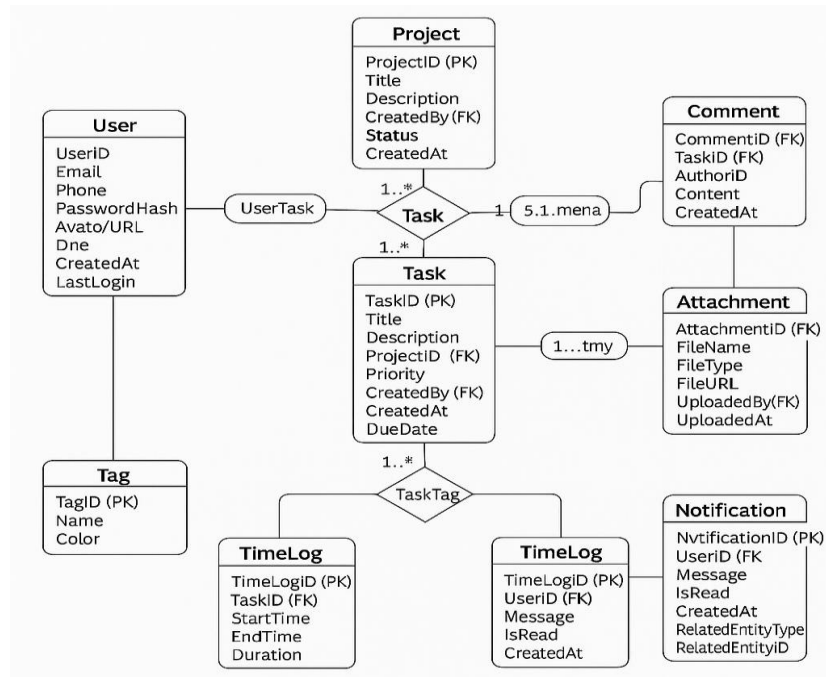


Рис. 4.1 Концептуальна модель БД

Логічна модель бази даних

Логічна модель деталізує таблиці, атрибути, ключі, типи зв'язків.

Таблиця 4.1

Users

Поле	Тип	Опис
user id (PK)	SERIAL	Унікальний ID користувача
role	VARCHAR(20)	employer / candidate
full_name	VARCHAR(100)	Ім'я та прізвище
email	VARCHAR(100)	Унікальний email
password_hash	TEXT	Хеш пароля
bio	TEXT	Опис профілю
avatar_url	TEXT	Фото
created_at	TIMESTAMP	Дата реєстрації

Таблиця 4.2

Companies

Поле	Тип
------	-----

company_id (PK)	SERIAL
user_id (FK)	INT
name	VARCHAR(150)
description	TEXT
website	VARCHAR(200)
logo_url	TEXT

Таблица 4.3

Vacancies

Поле	Тип
vacancy_id (PK)	SERIAL
user_id (FK)	INT
title	VARCHAR(200)
description	TEXT
salary_from	INT
salary_to	INT
location	VARCHAR(100)
created_at	TIMESTAMP

Таблица 4.4

Follows

Поле	Тип
follower_id (FK)	INT
following_id (FK)	INT
created_at	TIMESTAMP

Таблица 4.5

Applications

Поле	Тип
app_id (PK)	SERIAL
vacancy_id (FK)	INT
user_id (FK)	INT
resume_url	TEXT
cover_letter	TEXT
status	VARCHAR(20)
created_at	TIMESTAMP

Таблица 4.6

Chats

Поле	Тип
chat_id (PK)	SERIAL

user1_id (FK)	INT
user2_id (FK)	INT

Таблиця 4.7

Messages

Поле	Тип
message_id (PK)	SERIAL
chat_id (FK)	INT
sender_id (FK)	INT
text	TEXT
created_at	TIMESTAMP

Таблиця 4.8

Comments

Поле	Тип
comment_id (PK)	SERIAL
vacancy_id (FK)	INT
user_id (FK)	INT
text	TEXT
created_at	TIMESTAMP

Таблиця 4.9

Tags

Поле	Тип
tag_id (PK)	SERIAL
name	VARCHAR(50)

Таблиця 4.10

VacancyTags

Поле	Тип
vacancy_id (FK)	INT
tag_id (FK)	INT

Фізична модель бази даних

Фізична модель бази даних реалізована на основі реляційної СКБД PostgreSQL. У таблицях визначені типи даних, первинні та зовнішні ключі, обмеження цілісності, а також обмеження унікальності для забезпечення правильності збереження інформації.

Таблиця 4.11

Структура таблиці Users

Стовпець	Тип даних	Опис
user_id	SERIAL	Унікальний ідентифікатор користувача
role	VARCHAR(20)	Роль користувача: employer / candidate
full_name	VARCHAR(100)	ПІБ користувача
email	VARCHAR(100) UNIQUE	Унікальний email
password_hash	TEXT	Хеш пароля
bio	TEXT	Опис профілю
avatar_url	TEXT	Посилання на фото
created_at	TIMESTAMP	Дата створення облікового запису

Таблиця 4.12

Структура таблиці Companies

Стовпець	Тип даних	Опис
company_id	SERIAL	Унікальний ідентифікатор компанії
user_id	INT	Зовнішній ключ на Users
name	VARCHAR(150)	Назва компанії
description	TEXT	Опис діяльності
website	VARCHAR(200)	Офіційний сайт
logo_url	TEXT	Логотип

Таблиця 4.13

Структура таблиці Vacancies

Стовпець	Тип даних	Опис
vacancy_id	SERIAL	Унікальний ідентифікатор вакансії
user_id	INT	Зовнішній ключ на Users
title	VARCHAR(200)	Заголовок вакансії
description	TEXT	Опис вакансії
salary_from	INT	Мінімальна зарплата
salary_to	INT	Максимальна зарплата
location	VARCHAR(100)	Місто/країна
created_at	TIMESTAMP	Дата публікації

Таблиця 4.14

Структура таблиці Follows

Стовпець	Тип даних	Опис
follower_id	INT	Користувач, який підписується
following_id	INT	Користувач, на якого підписуються

created_at	TIMESTAMP	Дата створення підписки
------------	-----------	-------------------------

Таблиця 4.15

Структура таблиці Applications

Стовпець	Тип даних	Опис
app_id	SERIAL	Унікальний ідентифікатор відгуку
vacancy_id	INT	Зовнішній ключ на Vacancies
user_id	INT	Зовнішній ключ на Users
resume_url	TEXT	Посилання на резюме
cover_letter	TEXT	Супровідний лист
status	VARCHAR(20)	Статус (pending / accepted / rejected)
created_at	TIMESTAMP	Дата подачі

Таблиця 4.16

Структура таблиці Chats

Стовпець	Тип даних	Опис
chat_id	SERIAL	Унікальний ідентифікатор чату
user1_id	INT	Учасник чату
user2_id	INT	Другий учасник чату

Таблиця 4.17

Структура таблиці Messages

Стовпець	Тип даних	Опис
message_id	SERIAL	Унікальний ідентифікатор повідомлення
chat_id	INT	Зовнішній ключ на Chats
sender_id	INT	Користувач, що відправив повідомлення
text	TEXT	Текст повідомлення
created_at	TIMESTAMP	Час надсилання

Таблиця 4.18

Структура таблиці Comments

Стовпець	Тип даних	Опис
comment_id	SERIAL	Унікальний ідентифікатор коментаря
vacancy_id	INT	Зовнішній ключ на Vacancies
user_id	INT	Зовнішній ключ на Users
text	TEXT	Текст коментаря
created_at	TIMESTAMP	Дата публікації коментаря

Таблиця 4.19

Структура таблиці Tags

Стовпець	Тип даних	Опис
tag_id	SERIAL	Унікальний ідентифікатор тегу
name	VARCHAR(50)	Назва тегу

Таблиця 4.20

Структура таблиці VacancyTags

Стовпець	Тип даних	Опис
vacancy_id	INT	Зовнішній ключ на Vacancies
tag_id	INT	Зовнішній ключ на Tags

Таблиця 4.21

Структура таблиці ActivityLog (опціонально)

Стовпець	Тип даних	Опис
log_id	SERIAL	Унікальний запис
user_id	INT	Хто виконав дію
action	VARCHAR(50)	Тип дії (like, follow, apply, comment)
target_id	INT	ID об'єкта взаємодії
created_at	TIMESTAMP	Дата запису

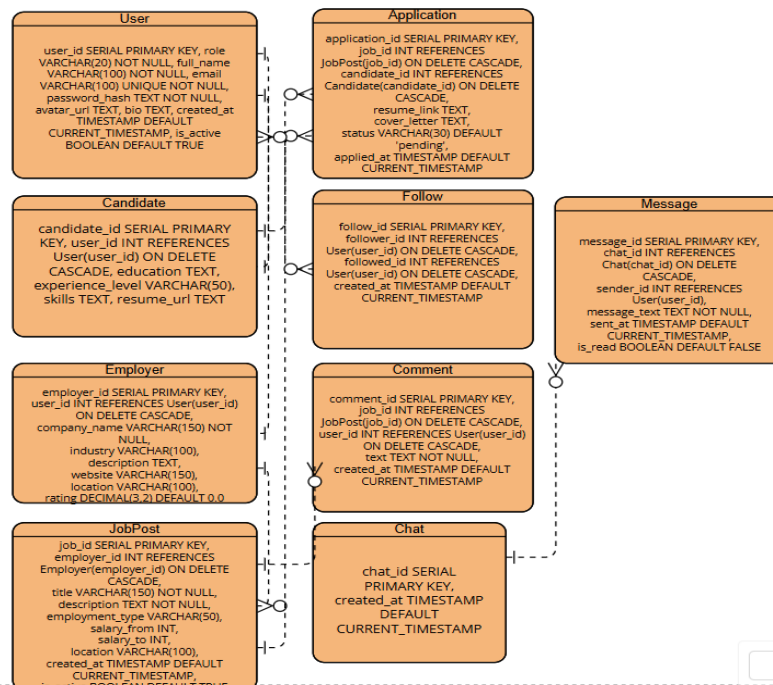


Рис. 4.2 Фізична модель БД

4.3 Розробка алгоритмів та інтерфейсів програмного забезпечення.

Ключові алгоритми

1. Формування стрічки вакансій
 - Алгоритм формує персоналізовану стрічку вакансій від підписаних профілів або загальнодоступних постів (аналог стрічки в Instagram).
 - Фільтрація за типом зайнятості, місцем роботи, тегами.
 2. Публікація вакансії (пост)
 - Схожа на створення посту: форма містить опис, заголовок, зарплатний діапазон, місце. Після публікації – відображається в стрічці підписників.
 3. Підписка на профілі
 - Реалізована як self-referencing зв'язок Follow. Користувач може підписатись на роботодавця або пошукача. Інформація про нові вакансії автоматично потрапляє в його стрічку.
 4. Коментарі до постів
 - Як у Instagram
 - До кожного посту/вакансії можна залишити коментар, побачити кількість відповідей, переглянути автора коментаря.
 5. Миттєві повідомлення (чат)
 - Через WebSocket. Інтерфейс нагадує DM в Instagram. Користувач може переглядати діалоги, надсилати/отримувати повідомлення в реальному часі[11].
 6. Відгуки на вакансії
 - Реалізовані у вигляді кнопки "Apply". Заявка надсилається з прикріпленим резюме (як файл або лінк), зберігається в базі.
- Ключові інтерфейси:
- Головна стрічка
 - Картки вакансій (із назвою, фото, описом, кнопкою "Apply", "Follow", "Comment").
 - Схожа за стилем на стрічку постів в Instagram.
 - Реалізовано нескінченне прокручування (infinite scroll).

- Профіль користувача
 - Як у Instagram: аватар, ім'я, біо, тип (роботодавець/пошукач), кількість підписок/підписників, всі опубліковані вакансії або історія відгуків.
- Сторінка вакансії
 - Детальний опис вакансії, кнопки "Apply", "Comment", можливість перегляду інших вакансій автора.
- Сторінка повідомлень
 - Список діалогів. Чат, як Instagram DM, з історією переписки та відображенням онлайн-статусу.
- Навігаційна панель
 - Іконки для: стрічки, пошуку, публікації, повідомлень, профілю (аналог нижнього меню Instagram).

За допомогою Figma було розроблено макет інтерфейсу відповідно до алгоритму (див рис. 4.2.1).

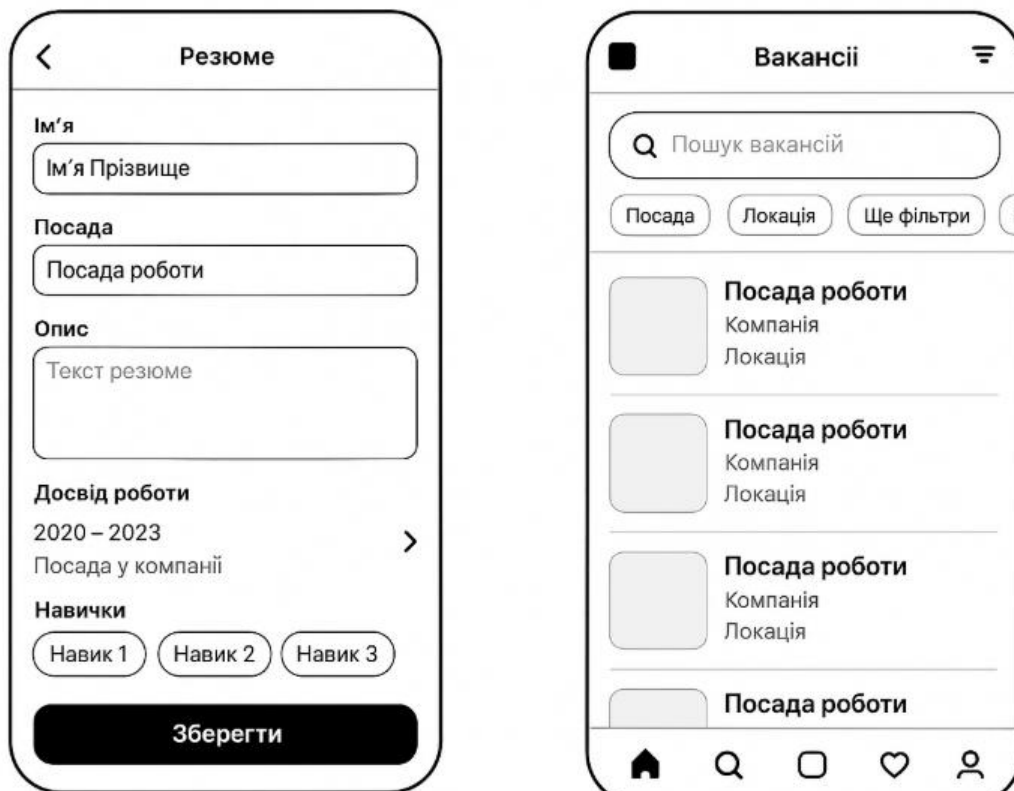


Рис 4.3 Макет інтерфейсу

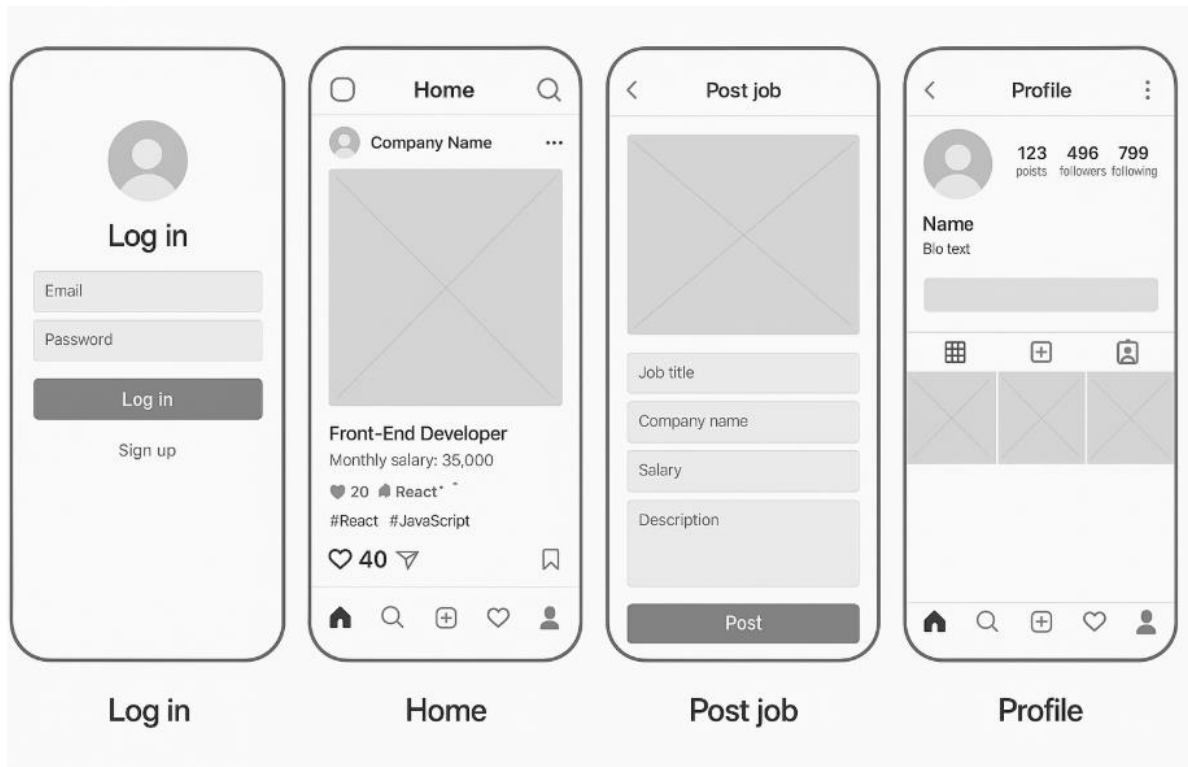


Рис 4.4 Макет інтерфейсу для входу

4.3 Тестування окремих функціональних елементів інформаційної системи

Тестування є обов'язковим етапом у життєвому циклі розробки веб-платформи. Його основна мета — виявити помилки на ранніх етапах, перевірити відповідність реалізованих функцій вимогам проекту, забезпечити стабільність і надійність системи під час реального використання[1].

Основні типи тестування, що застосовуються

1. Модульне тестування (unit testing)
Перевіряються окремі функції або компоненти (наприклад, обробка форми резюме, логіка ранжування кандидатів).
2. Інтеграційне тестування (integration testing)
Перевіряється взаємодія між модулями — наприклад, робота пошуку вакансій з базою даних, обмін повідомленнями між роботодавцем і шукачем.

3. Функціональне тестування (functional testing)

Аналізується, чи виконує система функції відповідно до вимог: подання заявки, реєстрація, пошук, рекомендації тощо.

4. Системне тестування (system testing)

Здійснюється перевірка всієї системи як єдиного цілого, в реальних умовах.

5. Тестування інтерфейсу (UI/UX testing)

Перевіряється коректність відображення інтерфейсів, відповідність дизайну, адаптивність для мобільних пристроїв.

6. Тестування безпеки та авторизації

Перевіряється захист особистих даних, доступ до функцій тільки для відповідних ролей (адміністратор, кандидат, роботодавець).

Інструменти, які можна використовувати

- Postman — тестування API (авторизація, запити до БД);
- Jest / Mocha (для JS) — модульне тестування клієнта/сервера;
- Selenium / Cypress — автоматизоване UI-тестування;
- SQLite / Moscaoo — для створення тестових даних;
- BrowserStack — кросбраузерне та адаптивне тестування.

Приклад проведення функціонального тестування

Функція: створення та збереження резюме.

Очікувана поведінка: після заповнення полів та натискання кнопки «Зберегти», дані мають зберігатись у БД та бути доступними у профілі.

Таблиця 4.22

Тестовий сценарій:

Крок	Вхідні дані	Очікуваний результат
1	Ім'я: Іван Іванов, Посада: Тестувальник	Поля заповнено
2	Натискання "Зберегти"	Переадресація у профіль
3	Перевірка у БД	Новий запис з'явився у таблиці Resumes

Методика покриття функціональних модулів тестами

Для отримання найкращих результатів варто застосовувати стратегію тестового покриття, коли кожен функціональний блок тестується щонайменше за такими сценаріями:

- Позитивний сценарій — правильне використання функції;
- Негативний сценарій — введення некоректних даних;
- Граничні значення — тестування на мінімальні/максимальні допустимі параметри;
- Тестування винятків — перевірка реакції системи на помилки (наприклад, відсутність з'єднання з БД).

Оцінка результатів тестування

Усі тести мають фіксуватися у тестовому звіті (тест-кейс), який включає:

- назву модуля;
- короткий опис тесту;
- вхідні дані;
- фактичний результат;
- статус: пройдено / не пройдено;
- коментар або виявлену помилку.

Таблиця 4.23

Тест-кейс приклад

№	Назва модуля	Вхідні дані	Очікувано	Отримано	Статус
1	Авторизація	логін, пароль	Вхід в систему	Успішно	Пройдено
2	Створення вакансії	незаповнене поле "опис"	Повідомлення про помилку	Не збережено	Пройдено

Якісне тестування — це основа надійності інформаційної системи. У межах веб-платформи для пошуку роботи тестування дозволяє забезпечити безперебійну взаємодію між шукачами та роботодавцями, запобігти технічним збоям та підвищити загальну зручність користування сервісом. Завдяки автоматизації тестування та покриттю основних функцій перевірками, система стає стабільною, масштабованою й готовою до реального використання.

4.4 Управління якістю

Забезпечення якості веб-платформи для пошуку роботи та підбору персоналу реалізується через комплексний підхід, що охоплює планування показників якості програмного продукту, контроль якості UX/UI-рішень на етапі проєктування інтерфейсу та перевірку відповідності системи функціональним вимогам. Така модель дозволяє забезпечити стабільність, безпеку та зручність використання платформи ще до початку повномасштабної розробки.

Для оцінки якості розроблюваної системи використовується міжнародний стандарт ISO/IEC 25010, який визначає ключові характеристики програмної якості. З огляду на специфіку проєкту — робота з персональними даними кандидатів, обмін повідомленнями, алгоритмічний підбір вакансій, веб-та мобільний інтерфейс з елементами соціальної взаємодії — основними характеристиками було обрано:

- Функціональну придатність (якість пошуку вакансій, відповідність резюме, рекомендаційний алгоритм)
- Безпеку (захист персональних даних, шифрування, безпечна автентифікація)
- Надійність (стабільність роботи сервера й чатів)
- Зручність використання (юзабіліті інтерфейсу, час виконання ключових дій)
- Продуктивність (швидкість пошуку, оптимізація стрічки вакансій)

- Сумісність (правильна робота на різних браузерях і мобільних пристроях)

Для забезпечення вимог стандарту сформовано план оцінки якості, наведений у таблиці 4.1

Таблиця 4.24

План оцінки якості продукту

Характеристика	Метрика	Метод збору даних	Критерій прийнятності
Безпека	Рівень шифрування даних	Аудит безпеки	AES-256 (зберігання), TLS 1.3 (передача)
Надійність	Доступність системи	Моніторинг серверів	$\geq 99.5\%$ доступності
Функціональність	Точність рекомендацій	Тестування алгоритму	Не менше 80% релевантності пропонуванних вакансій
Зручність використання	Час виконання ключової операції (відгук на вакансію)	Юзабіліті-тестування	≤ 60 секунд для нового користувача
Продуктивність	Час завантаження стрічки вакансій	Профілювання	≤ 2 секунд
Сумісність	Робота на ОС та браузерах	Тестування на Chrome, Safari, Firefox, Mobile	Коректна робота у $\geq 95\%$ тестів

Оцінювання якості прототипу

Оскільки проект знаходиться на етапі створення інтерактивного макета у Figma, фокус управління якістю спрямований на:

- повноту реалізації вимог у дизайні,
- зрозумілість навігації,
- зручність взаємодії з системою,
- доступність контенту,
- відповідність макету логіці реальних сценаріїв користування.

Для перевірки відповідності вимогам було створено **трасувальну матрицю**, яка пов'язує кожен функціональний вимогу з конкретним екраном та тест-кейсом.

Таблиця 4.25

Трасувальна матриця вимог проєкту

Вимога (ID)	Елемент інтерфейсу	Тест-кейс	Статус
FR001 (Реєстрація)	Екран Sign Up	ТС-01: Перевірка заповнення форми, унікальності email	Виконано
FR002 (Авторизація)	Login Screen	ТС-02: Вхід із правильними/неправильними даними	Виконано
FR003 (Створення профілю кандидата)	Candidate Profile	ТС-03: Завантаження фото, введення навичок, освіти	Виконано
FR004 (Створення вакансії)	Створення вакансії	ТС-04: Перевірка форми, обов'язкових полів	Виконано
FR005 (Пошук вакансій)	Екран Search & Filters	ТС-05: Тестування фільтрів (місто, зарплата, теги)	Виконано
FR006 (Подання заявки)	Vacancy Page	ТС-06: Відгук на вакансію, прикріплення файлу резюме	Виконано
FR007 (Підписки)	Profile → Followers	ТС-07: Підписка/відписка на роботодавця або кандидата	Виконано
FR008 (Рекомендації)	Recommendation Feed	ТС-08: Перевірка релевантності алгоритму	В макеті
FR009 (Чат)	Messages Screen	ТС-09: Надсилання та отримання повідомлень	В макеті
FR010 (Коментарі)	Vacancy → Comments	ТС-10: Додавання та видалення коментарів	Виконано
FR011 (Адмін-панель)	Admin Dashboard	ТС-11: Перевірка модерації коментарів / блокування аккаунтів	Заплановано

Оцінка зрілості процесів проєктування

Для визначення ефективності управління проектом було проведено самооцінку процесів життєвого циклу відповідно до ISO/IEC 33063. Оцінка здійснювалася за шкалою від 0 до 5, де 5 — повністю оптимізований процес.

Таблиця 4.26

Оцінка зрілості процесів проєктування

Процес	Мета	Рівень	Коментар
Управління вимогами	Узгодження функціоналу до завершення дизайну	4	Вимоги формалізовані у вигляді User Stories; є повний список FR/NFR
Валідація (дизайн)	Перевірка макетів на відповідність вимогам	2	Макети створені, але не всі екрани інтерактивні
Забезпечення якості (QA)	Контроль UX, перевірка доступності	2	Перевірено типографіку та контрасти; юзабіліті-тести ще не проводилися

Рекомендації з покращення процесів

Таблиця 4.27

План удосконалення процесів якості

№	Процес	Проблема	Рекомендація	Відповідальний / термін
1	Прототипування	Статичні макети не дають протестувати переходи	Створити інтерактивні зв'язки у Figma (User flow)	UI/UX Designer / 2 тижні
2	Передача в розробку	Немає UI Kit та стандартів компонентів	Створити UI Kit + документацію	UI/UX Designer / 3 тижні
3	Тестування UX	Потрібно оцінити зручність інтерфейсу	Провести юзабіліті-тестування на 5–7 користувачах	QA Engineer / 1 тиждень
4	SEO-перевірка	Немає перевірки продуктивності сторінок	Запустити аналіз у Google Lighthouse	Front-end / 3 дні

ВИСНОВОКИ

У ході виконання кваліфікаційної роботи на тему «Дослідження процесів управління проектом створення веб-платформи для майбутніх робітників та роботодавців» було досягнуто поставлену мету та послідовно реалізовано всі визначені завдання.

Обґрунтовано актуальність створення сучасного цифрового сервісу, який поєднує функціонал соціальної мережі та платформи для пошуку роботи. Проведені PEST- та SWOT-аналізи підтвердили високу доцільність реалізації проекту в умовах розвитку ринку дистанційної зайнятості, зростання попиту на швидкий пошук вакансій та переходу компаній до цифрових каналів рекрутингу. Визначено ключову конкурентну перевагу платформи — формування інтерактивного професійного профілю у форматі соціальної стрічки, що робить процес пошуку роботи інтуїтивнішим, візуальним і залученим для молодшої аудиторії.

Сформовано концептуальну модель системи, яка визначає взаємодію між підсистемами користувачів, роботодавців, рекомендаційного модуля, модуля вакансій і внутрішніх комунікацій. Побудовано математичну модель на основі теорії множин, яка формалізує структуру даних, логіку взаємодії між об'єктами та забезпечує чітке розуміння інформаційних потоків у системі.

Обґрунтовано вибір методології Scrum як найбільш гнучкої та такої, що відповідає умовам динамічних змін бізнес-вимог і потреб у швидкій реалізації MVP. Створено структурну декомпозицію робіт (WBS), яка охоплює весь життєвий цикл розробки. Сформовано організаційну структуру проекту на базі крос-функціональної Scrum-команди. Розроблено беклог продукту із деталізованими користувацькими історіями та критеріями приймання, що дало змогу перейти до планування спринтів та визначення пріоритетів розробки.

Проведено ідентифікацію ризиків, у ході якої проаналізовано потенційні загрози. За результатами експертного оцінювання було виділено найбільш критичні ризики, для яких розроблено превентивні та

коригувальні заходи. На основі оцінок обсягів робіт, навантаження команди та ринкових ставок виконано розрахунок бюджету проєкту, який включає резервний фонд для компенсації можливих фінансових загроз.

Розроблено архітектуру програмного забезпечення веб-платформи, що включає клієнтську частину, серверну логіку, інтеграції з зовнішніми сервісами, а також модулі реального часу (чат, стрічка оновлень). Спроектвано концептуальну, логічну та фізичну моделі реляційної бази даних, які забезпечують структуроване зберігання інформації про кандидатів, роботодавців, вакансії, відгуки та внутрішні повідомлення.

У створено макети ключових інтерфейсів платформи — стрічки вакансій, профілів кандидатів і роботодавців, системи відгуків, внутрішніх чатів, форми створення вакансій та пошуку з фільтрами.

Загалом отримані результати підтверджують ефективність застосовуваних методів проектного менеджменту та правильність обраної технологічної архітектури. Розроблена веб-платформа є технічно обґрунтованою, масштабованою, зручною у використанні й готовою до подальшої імплементації та розгортання MVP у рамках комерційного проєкту.

ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Agile Alliance. Agile Project Management Guide. – <https://www.agilealliance.org>
2. Beck K. Extreme Programming Explained: Embrace Change. – Addison-Wesley, 2005.
3. Boehm B. Software Engineering Economics. – Prentice Hall, 1981.
4. Bondarenko I.V., Ivashchenko I.Yu. Informatiini systemy i tekhnolohii v upravlinni personalom. – Kyiv: KNEU, 2020. – 224 s.
5. Coursera. Web Development Courses. – <https://www.coursera.org>
6. Figma. UI/UX Design Tool Documentation. – <https://www.figma.com>
7. GitHub. Open-source projects for job platforms. – <https://github.com>
8. Glass R.L. Software Engineering: A Practitioner’s Approach. – Addison-Wesley, 2002.
9. Google Developers. Web Performance Optimization. – <https://developers.google.com>
10. Instagram. Platform for visual content and social interaction. – <https://www.instagram.com>
11. ISO/IEC 12207:2017. Systems and software engineering — Software life cycle processes.
12. ISO/IEC 25010:2011. Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE).
13. Jooble. Global job search engine. – <https://jooble.org>
14. LinkedIn. Professional social network for employment and recruitment. – <https://www.linkedin.com>
15. Martin R.C. Clean Architecture: A Craftsman’s Guide to Software Structure and Design. – Prentice Hall, 2017.
16. McConnell S. Code Complete (2nd Edition). – Microsoft Press, 2004.
17. Mozilla Developer Network (MDN). Web Technologies Documentation. – <https://developer.mozilla.org>

18. Node.js Documentation. – <https://nodejs.org>
19. OECD. Digital Transformation and Employment. – <https://www.oecd.org>
20. Papazoglou M.P. Web Services: Principles and Technology. – Pearson Education, 2008.
21. PostgreSQL Documentation. – <https://www.postgresql.org>
22. Postman. API Testing and Development Tool. – <https://www.postman.com>
23. Pressman R.S. Software Engineering: A Practitioner’s Approach (8th Edition). – McGraw-Hill, 2014.
24. Rabota.ua. Employment platform in Ukraine. – <https://rabota.ua>
25. React.js Documentation. – <https://reactjs.org>
26. Robertson S., Robertson J. Mastering the Requirements Process. – Addison-Wesley, 2012.
27. Scrum.org. The Scrum Guide. – <https://www.scrumguides.org>
28. Sommerville I. Software Engineering (10th Edition). – Pearson Education, 2015.
29. Stack Overflow. Developer Survey 2023. – <https://survey.stackoverflow.co>
30. Statista. Employment and HR Digital Platforms Statistics. – <https://www.statista.com>
31. UX Collective. User Experience Design Articles. – <https://uxdesign.cc>
32. W3C. Web Content Accessibility Guidelines (WCAG) 2.1. – <https://www.w3.org/WAI/standards-guidelines/wcag/>
33. Work.ua. Ukrainian job search portal. – <https://www.work.ua>
34. World Economic Forum. The Future of Jobs Report. – <https://www.weforum.org>
35. Zhang P. Human–Computer Interaction and User Experience. – Springer, 2019.
36. Єфименко Т.І. Управління ІТ-проєктами. – Київ: КНЕУ, 2019.

37. Коваленко О.О. Проектування інформаційних систем. – Харків: ХНЕУ, 2018.
38. Костенко Н.В. Управління персоналом в умовах цифрової економіки. – Київ: КПІ ім. Ігоря Сікорського, 2021.
39. Лавріщева К.М. Інженерія програмного забезпечення. – Київ: Наукова думка, 2016.
40. Ситник Й.С. Інформаційні технології в HR-менеджменті. – Львів: ЛНУ ім. Івана Франка, 2020.

ДОДАТКИ

Додаток А

Таблиця А.1

Паспорт проекту мобільного додатку «Розробка проекту створення веб-платформи для майбутніх робітників та роботодавців»

№	Параметр	Опис
1	2	3
1	Назва проекту	Workgram. Розробка проекту створення веб-платформи для майбутніх робітників та роботодавців
2	Мета проекту	Створити веб-платформу з інтерфейсом, схожим на Instagram, для взаємодії студентів, випускників та молодих фахівців із потенційними роботодавцями. Платформа сприятиме спрощенню пошуку роботи, створенню особистих профілів та розміщенню вакансій у зручному, візуально привабливому форматі.
3	Завдання проекту	<ol style="list-style-type: none"> 1. Розробити функціонал реєстрації та аутентифікації користувачів (роботодавців і шукачів). 2. Створити особисті профілі у вигляді карток-портфоліо з фото, навичками, відео-презентаціями. 3. Забезпечити публікацію вакансій у форматі постів із тегами, описами, відео/зображеннями. 4. Реалізувати механізм відгуків на вакансії та особистих чатів. 5. Впровадити пошук і фільтрацію за спеціальністю, локацією, типом зайнятості. 6. Додати рекомендаційну систему вакансій на основі профілю користувача.
4	Цільова аудиторія	<ol style="list-style-type: none"> 1. Студенти та випускники вишів, технікумів, коледжів. 2. Молоді фахівці з досвідом до 3 років. 3. Роботодавці, що шукають молодих спеціалістів для стажування, практики або повної зайнятості.
5	Функціональні вимоги	<ol style="list-style-type: none"> 1. Створення профілю користувача з особистими даними, відео, фото, резюме. 2. Публікація вакансій із додаванням мультимедіа.

5	Функціональні вимоги	<ul style="list-style-type: none"> 3. Інтерактивна стрічка з вакансіями (аналог feed). 4. Система тегів та фільтрів для пошуку вакансій/кандидатів.
		<ul style="list-style-type: none"> 5. Месенджер для листування між роботодавцями та кандидатами. 6. Система відгуків, лайків, коментарів до вакансій/профілів. 7. Панель адміністратора для модерації контенту. 8. Налаштування приватності профілю.
6	Нефункціональні вимоги	<ul style="list-style-type: none"> 1. Веб-додаток з адаптивним дизайном (mobile-first). 2. Підтримка української, англійської мов. 3. Висока продуктивність (відгук інтерфейсу < 1 сек). 4. Безпека даних (шифрування, автентифікація через Google/Facebook). 5. Висока доступність (> 99.9%). 6. Відповідність принципам доступності WCAG 2.1. 7. Підтримка до 100 тис. одночасних користувачів. 8. Щоденні резервні копії.
7	Технічні характеристики	<ul style="list-style-type: none"> 1. - Фронтенд: React + TailwindCSS. 2. - Бекенд: Node.js (Express) з REST API. 3. - БД: PostgreSQL + Redis (для кешування). 4. - Хостинг: AWS (EC2, RDS, S3), Cloudflare для CDN і захисту. 5. - Інтеграції: Google OAuth, LinkedIn API (імпорт резюме), Telegram-боти для сповіщень. 6. - Месенджер: WebSocket для чату. 7. - UI/UX: Створено у Figma з урахуванням принципів Instagram UI. 8. - Безпека: HTTPS, JWT, обмеження прав доступу.

1	2	3
8	Життєвий цикл проєкту	<ol style="list-style-type: none"> 1. Дослідження та ТЗ (1 місяць): Аналіз аналогів (Work.ua, Rabota.ua, LinkedIn, Instagram, Jooble), формування вимог. 2. Проєктування (1,5 місяці): Архітектура, БД, UX/UI-дизайн. 3. Розробка MVP (3 місяці): Реєстрація, профілі, вакансії, чат. 4. Тестування (1,5 місяці): UI, навантажувальне, безпеки. 5. Запуск (1 місяць): Деплой на сервери, маркетинг. 6. Підтримка і масштабування (постійно): Збір зворотного зв'язку, релізи.
9	Очікувані результати	<ol style="list-style-type: none"> 1. Поліпшення якості підбору роботи для молоді. 2. Автоматизація процесу рекрутингу для компаній. 3. Мінімізація бар'єрів при працевлаштуванні студентів. 4. Очікуване зростання користувачів на 50 тис. за перший рік. 5. Вплив: цифровізація молодіжного ринку праці в Україні.
10	Організаційні аспекти	<ol style="list-style-type: none"> 1. Команда: Frontend та backend розробники, UX/UI дизайнер, DevOps, QA, менеджер проєкту, SMM. 2. Методологія: Scrum, щотижневі спринти. 3. Інструменти: Jira, GitHub, Slack, Notion, Figma. 4. Контроль якості: Code review, CI/CD, авто- та мануальне тестування.

**User Story Розробка проєкту створення веб-платформи для
майбутніх робітників та роботодавців»**

Код US	Формулювання US	Критерії прийняття
1	2	3
US001 (FR001)	Реєстрація. Як молодий фахівець або роботодавець, я хочу створити обліковий запис на платформі, щоб взаємодіяти з іншими та публікувати контент.	<ol style="list-style-type: none"> 1. Реєстрація через email або соціальні мережі (Google, LinkedIn). 2. Підтвердження email або номера телефону. 3. Обліковий запис зберігається в базі. 4. Користувач отримує повідомлення про успішну реєстрацію.
US002 (FR002)	Авторизація. Як зареєстрований користувач, я хочу входити у свій акаунт, щоб мати персоналізований доступ до функцій.	<ol style="list-style-type: none"> 1. Вхід через email/пароль або соцмережі. 2. Відображення повідомлення при помилці. 3. Доступна опція «забули пароль?» 4. Після входу відкривається персоналізована стрічка.
US003 (FR003)	Створення профілю. Як шукач роботи, я хочу заповнити профіль з фото, описом і відео, щоб привернути увагу роботодавців.	<ol style="list-style-type: none"> 1. Можна додати фото, резюме, відео-презентацію, навички, освіту. 2. Інтерфейс нагадує Instagram-профіль. 3. Можна редагувати профіль у будь-який час. 4. Дані зберігаються та відображаються публічно.
US004 (FR004)	Публікація постів. Як роботодавець або шукач, я хочу публікувати візуальні пости (вакансії, портфоліо), щоб бути поміченим.	<ol style="list-style-type: none"> 1. Публікація містить зображення, відео, опис, хештеги. 2. Пости відображаються у стрічці підписників.

US004 (FR004)	Публікація постів. Як роботодавець або шукач, я хочу публікувати візуальні пости (вакансії, портфоліо), щоб бути поміченим.	3. Можна лайкати, коментувати, ділитися. 4. Пости зберігаються в акаунті користувача.
US005 (FR005)	Пошук та фільтри. Як користувач, я хочу знаходити вакансії або кандидатів за фільтрами та тегами.	1. Доступна пошукова панель із фільтрами (локація, категорія, тип зайнятості). 2. Результати оновлюються в реальному часі. 3. Пошук працює з хештегами. 4. Підказки при введенні запиту.
US006 (FR006)	Підписки. Як користувач, я хочу підписуватись на інших, щоб бачити їхні нові публікації у своїй стрічці.	1. Кнопка «Підписатися» у профілях. 2. Нові пости відображаються у персональній стрічці. 3. Кількість підписників/підписок видно у профілі. 4. Можливість відписатися.
US007 (FR007)	Особисті повідомлення. Як роботодавець або кандидат, я хочу вести приватне листування для уточнення деталей.	1. Чат активується після взаємної взаємодії (відгук, підписка тощо). 2. Повідомлення зберігаються в історії. 3. Підтримка тексту, файлів, відео. 4. Сповіщення про нові повідомлення.
US008 (FR008)	Відгуки. Як роботодавець, я хочу залишити відгук про кандидата після співбесіди або роботи, щоб інші могли враховувати досвід.	1. Відгук доступний після взаємодії. 2. Можна поставити оцінку (1–5 зірок) і коментар. 3. Відгуки модеруються. 4. Середній рейтинг відображається у профілі.

US009 (FR009)	Сповіщення. Як користувач, я хочу отримувати повідомлення про події на платформі, щоб бути в курсі.	<ol style="list-style-type: none"> 1. Push-сповіщення про нові лайки, коментарі, повідомлення, відгуки. 2. Центр сповіщень у профілі. 3. Можливість налаштувати, які сповіщення отримувати.
US010 (FR010)	Аналітика профілю. Як користувач, я хочу бачити статистику переглядів і взаємодій з моїми постами та профілем.	<ol style="list-style-type: none"> 1. Відображення кількості переглядів, лайків, підписок. 2. Графіки взаємодій по днях/тижнях. 3. Можливість аналізувати популярність тегів. 4. Статистика доступна тільки власнику профілю.
US011 (FR011)	Система рекомендацій. Як шукач, я хочу бачити релевантні вакансії у стрічці на основі моїх уподобань.	<ol style="list-style-type: none"> 1. Алгоритм формує персональну стрічку за тегами, підписками, активністю. 2. Можна позначити вакансії як цікаві/нецікаві. 3. Чим більше активність – тим точніші рекомендації.
US012 (FR012)	Модерація контенту. Як адміністратор, я хочу мати змогу перевіряти та блокувати неприйнятний контент.	<ol style="list-style-type: none"> 1. Панель модератора з усіма скаргами. 2. Можливість блокування акаунтів або постів. 3. Логи дій адміністраторів зберігаються. 4. Користувачі отримують сповіщення про дії модераторів.

Беклог продукту проекту

Код US	Задача	Підзадача
US001	T01.1 Реєстрація користувача	ST01.1.1 Створення UI-інтерфейсу (email, соцмережі, пароль)
		ST01.1.2 Валідація даних
		ST01.1.3 Інтеграція OAuth 2.0 (Google, LinkedIn)
		ST01.1.4 Збереження профілю в базі
		ST01.1.5 Повідомлення про успішну реєстрацію
	T01.2 Вибір типу акаунта (роботодавець/пошукач)	ST01.2.1 Вивід варіантів під час реєстрації
		ST01.2.2 Призначення відповідної ролі
		ST01.2.3 Відображення персоналізованого інтерфейсу

US00 2	T02.1 Авторизація	ST02.1.1 Створення форми входу
		ST02.1.2 Перевірка правильності
		ST02.1.3 Повідомлення про помилку
		ST02.1.4 Перехід до стрічки
	T02.2 Відновлення пароля	ST02.2.1 Кнопка «забули пароль?»
		ST02.2.2 Відправка коду
		ST02.2.3 Встановлення нового пароля
US00 3	T03.1 Розробка профілю користувача	ST03.1.1 Форма заповнення (фото, опис, резюме, відео)
		ST03.1.2 Створення шаблону профілю як у соцмережах
		ST03.1.3 Збереження в базу даних
		ST03.1.4 Можливість редагування профілю

US00 4	T04.1 Створення постів	ST04.1.1 UI-додавання зображення/відео
		ST04.1.2 Поле опису, тегів, хештегів
		ST04.1.3 Збереження в БД, відображення в стрічці
	T04.2 Взаємодія з постами	ST04.2.1 Реалізація лайків і коментарів
		ST04.2.2 Відображення кількості взаємодій
		ST04.2.3 Повідомлення при новій активності
US00 5	T05.1 Пошук контенту	ST05.1.1 Пошукова панель із фільтрами
		ST05.1.2 Автодоповнення за хештегами
		ST05.1.3 Результати у вигляді сітки профілів/вакансій

US00 6	T06.1 Реалізація підписок	ST06.1.1 Кнопка «Підписатися»
		ST06.1.2 Взаємна підписка → активација чату
		ST06.1.3 Перелік підписників та підписок
US00 7	T07.1 Особисті повідомлення	ST07.1.1 Інтерфейс чату
		ST07.1.2 Збереження історії
		ST07.1.3 Повідомлення про новий меседж
		ST07.1.4 Підтримка прикріплення PDF, відео, CV
US00 8	T08.1 Відгуки	ST08.1.1 Форма відгуку (текст + зірки)
		ST08.1.2 Доступ після співпраці
		ST08.1.3 Модерація і публікація
		ST08.1.4 Рейтинг у профілі

US00 9	T09.1 Push-сповіщення	ST09.1.1 Події: новий фоловер, коментар, лайк
		ST09.1.2 Налаштування сповіщень
		ST09.1.3 Центр повідомлень у вебi
US01 0	T10.1 Аналітика профілю	ST10.1.1 Вивід переглядів, лайків, підписок
		ST10.1.2 Побудова графіка активності
		ST10.1.3 Доступ лише власнику профілю
US01 1	T11.1 Рекомендаційна стрічка	ST11.1.1 Алгоритм добору постів за підписками, тегами
		ST11.1.2 Реалізація пріоритетів у стрічці
		ST11.1.3 Налаштування інтересів
US01 2	T12.1 Адмін-модуль	ST12.1.1 Панель для перегляду контенту
		ST12.1.2 Можливість блокування постів/акаунтів
		ST12.1.3 Логи дій модераторів

SQL-скрипт створення структури бази даних веб-платформи

-- Таблиця користувачів

```
CREATE TABLE "User" (  
    user_id SERIAL PRIMARY KEY,  
    email VARCHAR(100) UNIQUE NOT NULL,  
    password_hash VARCHAR(255) NOT NULL,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- Таблиця профілів (пошукачі або роботодавці)

```
CREATE TABLE Profile (  
    profile_id SERIAL PRIMARY KEY,  
    user_id INT UNIQUE NOT NULL,  
    name VARCHAR(100) NOT NULL,  
    bio TEXT,  
    profile_type VARCHAR(20) CHECK (profile_type IN ('employer',  
'jobseeker')) NOT NULL,  
    avatar_url TEXT,  
    FOREIGN KEY (user_id) REFERENCES "User"(user_id) ON DELETE  
CASCADE  
);
```

-- Таблиця публікацій вакансій

```
CREATE TABLE JobPost (  
    post_id SERIAL PRIMARY KEY,  
    profile_id INT NOT NULL,  
    title VARCHAR(150) NOT NULL,  
    description TEXT NOT NULL,  
    location VARCHAR(100),  
    salary_range VARCHAR(50),
```

```
    post_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON DELETE  
CASCADE  
);
```

-- Таблиця заявок на вакансії

```
CREATE TABLE Application (  
    application_id SERIAL PRIMARY KEY,  
    post_id INT NOT NULL,  
    profile_id INT NOT NULL,  
    cover_letter TEXT,  
    status VARCHAR(20) CHECK (status IN ('pending', 'accepted', 'rejected'))  
DEFAULT 'pending',  
    apply_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (post_id) REFERENCES JobPost(post_id) ON DELETE  
CASCADE,  
    FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON DELETE  
CASCADE  
);
```

-- Таблиця підписок (self-referencing many-to-many)

```
CREATE TABLE Follow (  
    follower_id INT NOT NULL,  
    following_id INT NOT NULL,  
    follow_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    PRIMARY KEY (follower_id, following_id),  
    FOREIGN KEY (follower_id) REFERENCES Profile(profile_id) ON  
DELETE CASCADE,  
    FOREIGN KEY (following_id) REFERENCES Profile(profile_id) ON  
DELETE CASCADE  
);
```

-- Таблиця коментарів до постів

```
CREATE TABLE Comment (  
    comment_id SERIAL PRIMARY KEY,  
    post_id INT NOT NULL,  
    profile_id INT NOT NULL,  
    text TEXT NOT NULL,  
    comment_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (post_id) REFERENCES JobPost(post_id) ON DELETE  
CASCADE,  
    FOREIGN KEY (profile_id) REFERENCES Profile(profile_id) ON DELETE  
CASCADE  
);
```

-- Таблиця особистих повідомлень

```
CREATE TABLE Message (  
    message_id SERIAL PRIMARY KEY,  
    sender_id INT NOT NULL,  
    receiver_id INT NOT NULL,  
    message_text TEXT NOT NULL,  
    sent_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (sender_id) REFERENCES "User"(user_id) ON DELETE  
CASCADE,  
    FOREIGN KEY (receiver_id) REFERENCES "User"(user_id) ON DELETE  
CASCADE  
);
```